# U.S. Government Protection Profile for Web Servers in Basic Robustness Environments

Information
Assurance
Directorate

December 17, 2004

Version 0.61

## Table of Contents

## Table of Figures

## Table of Tables

# 1.0 Introduction

This U.S. Government Protection Profile for Web Servers in Basic Robustness Environments is sponsored by the National Security Agency (NSA) and is intended to be used as follows:

For product vendors and security product evaluators, this Protection Profile (PP) defines the requirements that must be addressed by specific products as documented in vendor Security Targets (STs).

For system integrators, this PP aids in identifying areas that need to be addressed to provide secure system solutions. By matching the PP with available STs, security gaps can be identified and products or procedures may be configured to bridge these gaps.

## *1.1 Identification*

This section provides information needed to identify and control this ST.

| | |
|---|---|
| Title: | U.S. Government Protection Profile for Web Servers in Basic Robustness Environments |
| Authors: | U.S. Government and industry |
| Vetting Status: | |
| CC Version: | 2.1 |
| Target EAL: | 2 Augmented |
| General Status: | |
| Registration: | |
| Keywords: | Web Server, HTTP, and HTTPS. |

## *1.2 Overview*

This PP specifies the minimum security requirements for all web servers containing National Security information operating in Basic Robustness Environments. Throughout this document, the web server may also be referred to as the Target of Evaluation (TOE). The target robustness level of "basic" is further discussed in Appendix D of this PP.

The TOE is a software application that serves content via a specific set of Internet protocols in response to requests from web users over a network. Some content is **public** and available to any requestor; other content is **controlled-access** and must be protected from unauthorized disclosure. Determination of whether content is public or controlled, and the information contained in the content, is under the control of a **content provider**. The TOE must prevent users from modifying content and minimize the risk of malicious code from modifying content. A complete description of the TOE may be found in Section 2.0 of this PP.

This PP defines:

assumptions about the security aspects of the environment in which the TOE will be used;

threats that are to be addressed by the TOE and its IT environment;

security objectives of the TOE and its environment;

functional and assurance requirements to meet those security objectives; and

rationale demonstrating how the requirements meet the security objectives, and how the security objectives address the threats.

The protection profiles related to this PP fall into three categories:

Interfacing Protection Profiles. These PPs define the security requirements for applications that interface with the Web Server. This includes the Web Browser Protection Profile, which provides the security requirements to support the end-user interface to the web server, as well as a Web Application Protection Profile, which define the security requirements for executable web content.

Application Protection Profiles. These PPs define the security requirements for other networking applications that can directly or indirectly interface with the Web Server, such as servers for other Internet protocols.

Platform Protection Profiles. These PPs define appropriate security requirements for underlying platforms. This includes the Controlled Access Protection Profiles (CAPP), as well as other Operating System Protection Profiles that provide basic or stronger robustness.

### 1.2.1 Environmental Characterization

The TOE is expected to be executing on an operating system that has successfully passed an evaluation against a NSA approved operating system PP at a basic robustness level or higher. The hardware is expected to be physically protected to a level commensurate with the data it processes. Only authorized administrators are allowed physical access to the server, are cleared to the level of the information being served and possess a need-to-know of the information being served.

All hosts able to connect to the TOE are approved to process information at the highest level served by the TOE, but users may not possess a need-to-know all of the information served by the TOE. Therefore the TOE is in a benign environment.

## *1.3 Conventions*

The notation, formatting, and conventions used in this PP are largely consistent with those used in Version 2.1 of the Common Criteria (CC). Selected presentation choices are discussed here to aid the PP user.

The CC allows several operations to be performed on functional requirements; *refinement*, *selection*, *assignment*, and *iteration* are defined in paragraph 2.1.4 of Part 2 of the CC. Each of these operations is used in this PP.

The **assignment** operation is used to assign a specific value to an unspecified parameter, such as the length of a password. The **selection** operation is used to select one or more options provided by the CC in stating a requirement. Completed assignment and selection operations are denoted by *italicized text*.

**Iteration** of a component is required when an operation within the component must be completed multiple times with differing values, or for different allocation of functions to partitions of the TOE. Iterated functional and/or assurance components are given unique identifiers by appending a slash ("/") and an iteration identifier to the element identifiers from the CC. (e.g. FDP_ACF.1.1/CP, FDP_ACF.1.2/CP)

The **refinement** operation is used to provide an elaboration of an existing CC element to explicitly meet stated objectives. Refinement of elements is denoted by **bold text**.

Application notes document guidance for how the requirement to be applied. Rather than being collected into a separate section, the application notes are integrated with requirements and indicated as notes. Application notes should be considered informative.

In the requirement sections, each section that represents a requirement family or component, there is a mnemonic in parenthesis. These refer to the requirement section in the CC from which it was derived. Requirement elements have these references includes as superscripted text at the end of the element.

## 1.4 Glossary

A glossary has been included in Appendix B.

## 1.5 Organization of this Document

Section 1.0, Introduction, provides document management and overview information necessary to identify the PP along with references to other related PP's.

Section 2.0, TOE Description, defines the TOE and establishes the context of the TOE by referencing generalized security requirements.

Section 3.0, Security Environment, describes the expected environment in which the TOE is to be used. This section defines the set of threats that are relevant to the secure operation of the TOE, organizational security policies with which the TOE must comply, and secure usage assumptions applicable to this analysis.

Section 4.0, Security Objectives, defines the set of security objectives to be satisfied by the TOE and by the TOE operating environment.

Section 5.0, IT Security Requirements, defines the security functional and assurance requirements derived from the Common Criteria, Part 2 and Part 3, respectively, that must be satisfied by the TOE and the Non-IT environment.

Section 6.0, Rationale, provides rationale to demonstrate that the security objectives satisfy the threats and policies. This section also explains how the set of requirements are complete relative to the security objectives and presents a set of arguments that address dependency analysis and Strength of Function (SOF) and use of the explicit requirement.

Section 7.0, Appendices, provides a list of references, a glossary, acronyms and a discussion of Robustness.

# 2.0 TOE Description

## 2.1 Product Type

This protection profile provides an implementation independent specification of the security functional requirements for a web server for use in National Security Systems operating in a basic robustness environment. The web server is an application program running on a operating system and hardware platform. It is assumed that the operating system and underlying hardware have been previously evaluated against a basic robustness or higher PP for use in National Security Systems and that the operating system provides:

identification and authentication,

discretionary access controls,

process isolation, and

audit functions.

The web server is able to serve both **static** and **dynamic** content using Hypertext Transfer Protocol (HTTP) and HTTP over Secure Socket Layer (HTTPS). The content served represents information provided by a **content provider** to a web user. Static content is provided to the web user ‘as is’, with no processing performed by the web server (i.e., HTML, Java, JavaScript). Dynamic content is content that is generated on the fly, either being assembled by the server or as the output of executable content.

Some content is **public content**, which means that it is available to any web user that requests it without authentication. Other content is **controlled access content**, which means that the content is distributed only to web users authorized for that content by the content provider. Note that each content provider has control over the sets of web users authorized to access their content. If the web server is used in a classified environment, it is assumed that all users are cleared to the level of the information being served, however all users do not necessarily have a need-to-know.

## 2.2 TOE Definition

The TOE is a web or HyperText Transport Protocol (HTTP) server designed to receive requests for information (content) and deliver that information to the requester. HTTP servers were originally designed to receive anonymous requests from unauthenticated hosts on the Internet. However, HTTP servers have evolved to deliver restricted information through the same common client interface (a “brower”) and referenced by a Universal Resource Locator (URL).

Web servers compliant with this PP provide support for encryption through the SSL and TLS protocols. While most browsers are able to handle many different protocols (e.g. FTP, TELENT, NEWS etc.), the security of non-HTTP protocols is not directly addressed by this profile; instead, they are addressed by specific protection profiles for each type of server.

For the purposes of this protection profile, **web servers** are application programs. They execute on a host platform that provides the underlying abstractions used to store content and execute programs. The web server controls access to information by the use of its own security features in combination with the features provided by the host platform.

## 2.3 TOE functionality

The TOE responds to requests for public information using the HyperText Transport Protocol (HTTP). The content provided can reside in static files (e.g. HTML files) or more dynamic content can be generated "on-the-fly" (e.g. Common Gateway Interface, Active Server Pages, Java Server Pages etc.). These web applications that create content on-the-fly are beyond the scope of this PP and are addressed by the Web Application Protection Profiles.

The TOE is also able to deliver restricted content using HTTP (secure) also referred to as HTTPS using FIPS 140-2 validated SSL v3.0 or TLS v1.0. Identification and authentication of web users can be provided through personal digital certificates or through user ID and password schemes[1].

While HTTP is an extensible protocol, the standard (RFC 2616) defines the following eight methods that can performed on the resource identified by the requested Universal Resource Identifier (URI): OPTIONS, GET, HEAD, POST, PUT, DELETE, TRACE and CONNECT. The ST must address any additional methods supported by the TOE in a manner consistent with the objectives defined in this PP.

Figure 2-1 provides the conceptual model of the TOE's placement in an overall network. Alternately, multiple forms of network application services (web server, FTP server, terminal server) could be located on the same machine. The key point, applicable to the services, is that the operating system provides low-level mediation of access to files.

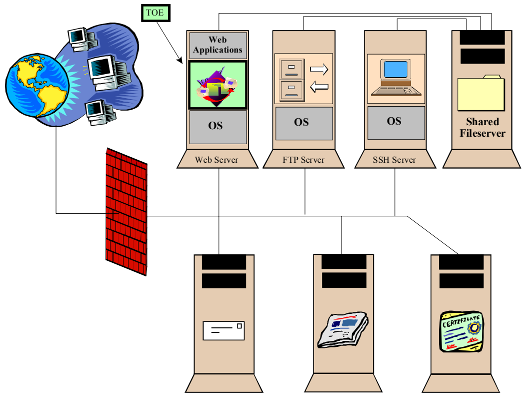

**Figure 2-1: Placement of the TOE in an overall System Architecture**

[1] The TOE may also provide support for password protection and the serving of password protected content over unencrypted connections, but such support is not a secure usage for protected data, and is assumed not to be used by those who consider their data controlled access.

## 2.4 TOE Operational Environment

The TOE is expected to be executing on an OS and hardware platform that have been evaluated against a NIAP validated (basic robustness or higher) operating system PP providing the following security functions:

- identification and authentication,
- discretionary access controls,
- process isolation, and
- audit functions.

The physical web server is expected to be physically protected to a level commensurate with the data it processes. Only authorized administrators are allowed physical access to the server. All users and administrators are cleared to the level of the information being served but some users may not possess a need-to-know of the information being served.

The administrator establishes the configuration of the server, and controls the set of authorized content providers. To secure the content provided by the TOE, the administrator and content providers have the capability to control the access of web users.

## 2.5 Security Function Policies (SFPs)

TOE evaluation is concerned primarily with ensuring that a defined TOE Security Policy (TSP) is enforced over the TOE resources. The TSP defines the rules by which the TOE governs access to its resources, and thus all information and services controlled by the TOE.

The TSP is, in turn, made up of multiple Security Function Policies (SFPs). Each SFP has a scope of control, that defines the subjects, objects, and operations controlled under the SFP. The SFP is implemented by a Security Function (SF), whose mechanisms enforce the policy and provide necessary capabilities.

Because this basic robustness PP is intended for the evaluation of application web servers running on a PP compliant operating system, it is necessary to describe the SFPs of the TOE and those of the environment (OS) which are necessary for the correct operation of the TOE.

The following paragraphs describe the security function policies (SFPs) used in this PP.

### 2.5.1 TOE security function policies

WEBUSER (WU) SFP

The intent of the WEBUSER SFP is to control access by entities accessing the server over the network to obtain content. All other operations between these subjects and objects are expressly denied. The WEBUSER SFP is summarized in the following table:

**Table 2-1: A summary of the WEBUSER SFP**

| Subject[2] | Object | Operation[3] | Description |
|---|---|---|---|
| User | Public content | Read | Any user may access any public content provided by the TOE. |
| User | Controlled-access content | Read | To access controlled content, an authorized user must be authenticated and the access must be explicitly permitted by the content provider. |

### 2.5.2 Environmental security function policies

Content-Provider (CP) SFP

The Content-Provider (CP) SFP dictates the rules that control the ability for content providers (typically, a subset of the users on the host platform) to install and modify content. Unlike typical DAC (discretionary access control) policies, this SFP is more centrally controlled, with the TOE administrator having control over the ability of the content providers to install and modify content.

**Table 2-2: A summary of the CONTENT-PROVIDER SFP**

| Subject[4] | Object | Operation | Description |
|---|---|---|---|
| Content provider | Public content | Read, write | Each content provider is permitted control over the management of their content. |
| Content provider | Controlled access content | Read, write | To access controlled content, a content provider must be authenticated. |

## *2.6 Use of this PP*

This PP is intended to identify the minimum security features of a web server in basic robustness environments. The underlying hardware and operating system are specifically excluded from the TOE. As a result, some web servers will rely on the host operating environment to manage content providers and to provide tools to create, modify, and manage content.

[2] A subject is a process acting on behalf of the entity specified.

[3] The only operation permitted by this SFP is the read operation. Other operations (e.g. delete, modify, rename) that may exist are outside of the scope of the TOE

[4] A subject is a process acting on behalf of the entity specified.

There are products that also manage content providers and provide tools to create, modify, and manage content directly through the content provider's browser. These products can also show conformance to this PP by reallocating policies, objectives and SFRs from the environment to the TOE.

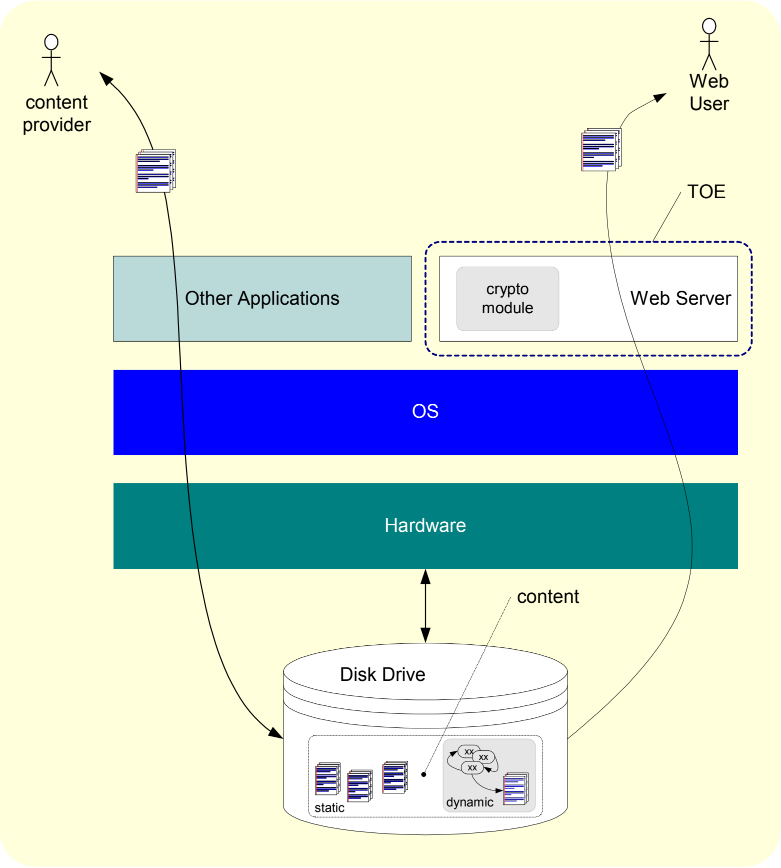

**Figure 2-2: The TOE and its execution environment**

# 3.0 Security Environment

Basic robustness TOEs fall in the upper left area of the grids shown in Figure 7-1 and Figure 7-2. Basic robustness is considered sufficient for low threat environments or where compromise of protected information will not have a significant impact on mission objectives. This implies that the motivation of the threat agents will be low in environments that are suitable for basic robustness. In general, basic robustness results in "good commercial practices" that counter threats based in casual and accidental disclosure or compromise of data protected by the TOE.

Threat agent motivation can be considered in a variety of ways. One possibility is that the value of the data process or protected by the TOE will generally be seen as of little value to the adversary (i.e., compromise will have little or no impact on mission objectives). Another possibility, (where higher value data is processed or protected by the TOE) is that procuring organizations will provide other controls or safeguards (i.e., controls that the TOE itself does not enforce) in the fielded system in order to increase the threat agent motivation level for compromise beyond a level of what is considered reasonable or expected to be applied.

## *3.1 Threats*

The following sections provide a characterization of the threat agent and describe the threats addressed by the TOE. Since this PP covers software only web servers used in Basic Robustness Environments, the host operating system will be required to provide security functions for the TOE. The threats against the operating system are also described.

### 3.1.1 Threat Agent Characterization

Section 7.4 contains an in-depth discussion of threat characterization for basic robustness environments.

### 3.1.2 Threats countered or partially countered by the TOE

T.CAPTURE_TRAFFIC A web user may attempt to access non-public content by reading TCP/IP datagrams directly "off the wire" using a network traffic analyzer (e.g. "sniffer", packet analyzer, etc.) or a "man-in-the-middle" attack.

T.INVALID_URL A web user may attempt to create, modify or view controlled-access content, web server configuration files or OS specific files by entering an invalid URL or a URL specifically designed for this purpose.

T.MASQUERADE A user may masquerade or replay a previous session of another web user in order to access controlled content that would not normally be accessible.

T.SERVER_MASQ A user may attempt to masquerade his web server as the legitimate web server to provide false or misleading content or capture user data.

T.UNAUTHORIZED A web user may request controlled-access content for which they are not authorized.

### 3.1.3 Threats countered by the Environment

$T_e$.PROVIDER_MASQ A user may attempt to create, modify or delete content that they are not authorized to by masquerading as the proper content provider.

$T_e$.REPLAY A user may attempt to masquerade as the administrator by capturing and replaying valid identification and authentication information.

$T_e$.TSF_BYPASS A user may attempt to bypass the TSF to create, modify or delete controlled-access content, TSF data or other OS configuration files or the TOE by using non TOE interfaces of the host computer system.

## 3.2 Organizational Security Policies

Organizational security policy statements are statements of the rules, practices or guidelines that must be followed by the TOE or its environment, as determined by the organization controlling the environment in which the TOE is to be used. An example organizational security policy is a requirement for password generation and encryption to conform to a standard stipulated by a national government.

PP-compliant TOEs must address the organizational security policies described below.

P.CRY_APM Any cryptographic-based security must use NIST-approved algorithms.

P.CRY_VAL Any cryptographic-based security components used to protect sensitive information on U.S. Government computer must be FIPS 140-2 validated.

P.SYS_BNR Each computer system will display restrictions of use, legal agreements or any other appropriate information to which users consent by accessing the system.

P.USR_ACC The users of the TOE will be held accountable for their actions within the TOE.

## 3.3 Assumptions

This section describes assumptions used to prepare this protection profile. These assumptions cover aspects of physical and personnel security; as well as connectivity of the TOE and its environment.

A.ADM_GOOD Administrators will follow all published guidance.

A.ADM_TRND Administrators will be appropriately trained.

A.ADM_TRSTD Administrators will not intentionally attempt to violate the TOE security policy or any environmental security policies necessary for the correct operation of the TOE.

$A_{os}$.PHY_ACCES Physical access to the host computer system will be restricted to authorized personnel.

$A_{os}$.PHY_PROT Physical protection of the host computer system will be commensurate with the value of that computer system and the data it contains.

$A_{ws}$.CPR_EAC Content providers will establish access controls in accordance with the handling and dissemination procedures for that content.

$A_{ws}$.CPR_GOOD Content providers will follow all published guidance.

$A_{ws}$.CPR_TRND Content providers will be trained on the handling and dissemination procedures for the content for which they are responsible.

$A_{ws}$.CPR_TRSTD Content providers will not intentionally attempt to violate the TOE security policy or any environmental security policies necessary for the correct operation of the TOE.

$A_{ws}$.SYS_HIGH All users with access to the host computer system possess proper personnel security clearance for all data contained on that system but only selected users or groups of users may obtain access to that data (e.g., based on a need-to-know).

# 4.0 Security Objectives

This chapter describes the security objectives for the TOE and the TOE's operating environment. The security objectives are divided between TOE Security Objectives (i.e., security objectives addressed directly by the TOE) and Security Objectives for the Operating Environment (i.e., security objectives addressed by the IT domain or by non-technical or procedural means).

## 4.1 Security Objectives for the Web Server

This section defines the security objectives that are to be addressed by the Web Server.

$O_{ws}$.SYS_PROT The web server will protect TOE data and content from unauthorized modification, deletion or disclosure.

$O_{ws}$.AUD_GEN The web server will detect security relevant events and create a protected record of these events.

$O_{ws}$.SSL_TLS The web server will support SSL v3.0/TLS v1.0 or higher.

## 4.2 Security Objectives for the Web Server Cryptographic Module

This section defines the security objectives that are to be addressed by the cryptomodule.

$O_{cm}$.CMVP Cryptographic mechanisms will be NIST FIPS 140-2 validated.

$O_{cm}$.COP_AMD Cryptographic mechanisms will default to NIST FIPS140-2 approved functions and modes of operation.

$O_{cm}$.COP_SFT A self-test will ensure the correct operation of cryptographic mechanisms.

## 4.3 Security Objectives for the operating system

This section defines the security objectives that are to be addressed by the TOE.

$O_{os}$.AUD_FUN The operating system will keep a protected record of security relevant events and allow administrators to easily use this record to investigate security incidents. Application programs must have the ability to submit events to this record.

$O_{os}$.I&A The operating system will provide the ability to uniquely identify and authenticate the administrators and content providers.

$O_{os}$.SYS_BNR The operating system will provide a banner that describes the restrictions of use, legal agreements etc. that the user must agree to prior to proceeding with the session.

$O_{os}$.SYS_PROT The operating system will protect itself, the TOE, TOE data and content from unauthorized modification, deletion or disclosure.

### *4.4 Non-IT Security Objectives for the environment*

$O_E$.ADMIN A process exists for the hiring and training of qualified personnel that can be trusted to handle the all the content provided by the TOE.

$O_E$.CON_PROV A process exists for the hiring and training of qualified personnel that can be trusted to manage content dissemination.

$O_E$.PROT The environment will provide physical protection and access controls such that only authorized personnel are permitted access to the host computer system the TOE is running on.

$O_E$.SYS_HIGH The environment will provide physical controls such that access (physical and logical) to the network served by the web server is restricted to those personnel that possess proper personnel security clearance for all data contained on that system.

# 5.0 IT Security Requirements

This section provides functional and assurance requirements for the TOE and the host operating system that must be satisfied by a PP-compliant solution. These requirements consist of functional components from Part 2 of the CC and assurance requirements from Part 3.

## 5.1 TOE Security Functional Requirements

The functional security requirements for the TOE consist of the following components derived from Part 2 of the CC.

### 5.1.1 Web Server Functional Security Requirements

**Table 5-1: Web Server Security Functional Requirements**

| Identifier | Description |
|---|---|
| FAU: Security Audit | |
| FAU_GEN.1-NIAP-0410 | Audit data generation |
| FAU_GEN.2-NIAP-0410 | User identity association |
| FDP: User Data Protection | |
| FDP_ACC.1/WU | Subset Access Control (SFP WEBUSER) |
| FDP_ACF.1-NIAP-0407/WU | Security Attribute Based Access Control (SFP: WEBUSER) |
| FDP_RIP.1 | Subset Residual Information Protection |
| FDP_UCT.1/WU | Basic Data Exchange Confidentiality (SFP: WEBUSER) |
| FDP_UIT.1/WU | Data Exchange Integrity (SFP: WEBUSER) |
| FIA: Identification and authentication | |
| FIA_AFL.1-NIAP-0425 | Authentication failure handling |
| FIA_ATD.1 | User Attribute Definition |
| FIA_UAU.1 | Timing of Authentication |
| FIA_UID.1 | Timing of Identification |
| FIA_USB.1-NIAP-0351 | User-Subject Binding |
| FMT: Security management | |
| FMT_MOF.1 | Management of Security Functions Behavior |
| FMT_MSA.1 | Management of security attributes |
| FMT_MSA.2 | Secure Security Attributes |
| FMT_MSA.3-NIAP-0429 | Static attribute initialization |
| FMT_MTD.1 | Management of TSF Data |
| FMT_REV.1 | Revocation |
| FMT_SMF.1 | Specification of Management Functions |
| FMT_SMR.1 | Security Roles |

#### 5.1.1.1 FAU: Security Audit

FAU_GEN.1-NIAP-0410: Audit data generation

Hierarchical to: No other components.

FAU_GEN.1.1-NIAP-0410 The TSF shall be able to generate an audit record of the following auditable events:

a) Start-up and shutdown of the audit functions;

b) [selection: [assignment: *events at a basic level of audit introduced by the inclusion of additional SFRs determined by the ST author*], [assignment: *events commensurate with a basic level of audit introduced by the inclusion of explicit requirements determined by the ST author*], "*no additional events*"].

c) All auditable events listed in Table 5-2: Auditable Events, below.

Application Note

FAU_GEN.1.1-NNIAP-0410.a it is sufficient for the TOE to post an audit record when it begins capturing audit events and post another audit record when it terminates cleanly. It is recognized that he TOE is an application and will not directly control the audit functions provided by the OS.

FAU_GEN.1.2-NIAP-0410 The TSF shall record within each audit record at least the following information:

a) Date and time of the event, type of event, subject identity (if applicable), and the outcome (success or failure) of the event; and

b) For each audit event type, based on the auditable event definitions of the functional components included in the PP/ST, the information specified in column three of Table 5-2: Auditable Events below.

Dependencies: FPT_STM.1 Reliable time stamps

Application Notes:

For the selection, the ST author should choose one or both of the assignments (as detailed in the following paragraphs), or select "no additional events".

For the first assignment, the ST author should augment the table (or lists explicitly) the audit events associated with the basic level of audit for any SFRs that the ST author includes that are not included in this PP.

For the second assignment the ST author should include audit events that may arise due to the inclusion of any explicit requirements not already in the PP. Because "basic" audit is not defined for such requirements, the ST author will need to determine a set of events that are commensurate with the type of information that is captured at the basic level for similar requirements.

If no additional (CC or explicit) SFRs are included, or if additional SFRs are included that do not have "basic" audit associated with them, then it is acceptable to assign "no additional events" in this item.

In column 3 of Table 5-2, "if applicable" is used to designate data that should be included in the audit record if it "makes sense" in the context of the event that generates the record. If no other information is required (other than that listed in "a") for a particular audit event type, then an assignment of "none" is acceptable.

**Table 5-2: Auditable Events**

| Requirement | Auditable Events | Additional Audit Record Contents |
|---|---|---|
| FAU_GEN.1-NIAP-0410 | None | None |
| FAU_GEN.2-NIAP-0410 | None | None |
| FDP_ACC.1/WU | None | None |
| FDP_ACF.1-NIAP-0407/WU | All requests to perform an operation on an object covered by the SFP. | None |
| FDP_RIP.1 | None | None |
| FDP_UCT.1/WU | Unauthorized user attempting to use the data exchange mechanisms. | A reference to the names or other indexing information useful in identifying the user data that was transmitted or received. |
| FDP_UIT.1/WU | Unauthorized user attempting to use the user data exchange mechanisms | A reference to the names or other indexing information useful in identifying the user data that was transmitted or received. Any identified attempts to block transmission of user data. |
| FIA_AFL.1-NIAP-0425 | The reaching of the threshold for the unsuccessful authentication attempts. | The actions taken and the subsequent, if appropriate, restoration to the normal state |
| FIA_ATD.1 | None | None |
| FIA_UAU.1 | All use of the authentication mechanism. | None |
| FIA_UID.1 | All use of the user identification mechanism, including the user identity provided. | None |
| FIA_USB.1-NIAP-0351 | Success and failure of binding of user security | None |

| Requirement | Auditable Events | Additional Audit Record Contents |
|---|---|---|
| | attributes to a subject (e.g. success and failure to create a subject). | |
| FMT_MOF.1 | All modifications in the behavior of the functions in the TSF. | None |
| FMT_MSA.1 | All modifications of the values of security attributes. | None |
| FMT_MSA.2 | All offered and accepted secure values for a security attribute. | None |
| FMT_MSA.3-NIAP-0429 | Modifications of the default setting of permissive or restrictive rules.<br>All modifications of the initial values of security attributes. | None |
| FMT_MTD.1 | All modifications to the values of TSF data. | None |
| FMT_REV.1 | All attempts to revoke security attributes. | None |
| FMT_SMF.1 | Use of the management functions. | None |
| FMT_SMR.1 | None | None |

FAU_GEN.2-NIAP-0410: User identity association

Hierarchical to: No other components.

FAU_GEN.2.1-NIAP-0410 For audit events resulting from actions of identified users, the TSF shall be able to associate each auditable event with the identity of the user that caused the event.

Dependencies: FAU_GEN.1 Audit data generation
FIA_UID.1 Timing of identification

### 5.1.1.2 FDP: User Data Protection

FDP_ACC.1/WU: Subset Access Control/WU

Hierarchical to: No other components.

FDP_ACC.1.1/WU The TSF shall enforce the *WEBUSER SFP* on

a) Subject: Process acting on behalf of a web user

b) Objects: Controlled-access content

c) Operations

1. GET
2. PUT
3. DELETE

Dependencies: FDP_ACF.1 Security attribute based access control

FDP_ACF.1-NIAP-0407/WU: Security Attribute Based Access Control/WU

Hierarchical to: No other components

FDP_ACF.1.1-NIAP-0407/WU The TSF shall enforce the *WEBUSER SFP* to objects based on the following:

*a*) Subject attributes:

1. username

b) Object attributes:

1. Realm,
2. [selection: *content identifier (e.g. file name),*
3. [assignment: *unique content identifier*]]

FDP_ACF.1.2-NIAP-0407/WU The TSF shall enforce the following rules to determine if an operation among controlled subjects and controlled objects is allowed:

a) If the requested operation is PUT then deny the operation;

b) If the requested operation is DELETE then deny the operation;

c) If the object is *public content* then permit the GET operation.

d) If the object is *controlled-access content* and the authenticated user's username is associated with the realm containing the object then permit the operation.

d) Otherwise, deny the operation.

FDP_ACF.1.3-NIAP-0407/WU The TSF shall explicitly authorize access of subjects to objects based on the following additional *WEBUSER SFP* rules:

a) [selection: [assignment: *rules, based on security attributes, that explicitly authorize access of subjects to objects*], "no additional rules"]

FDP_ACF.1.4-NIAP-0407/WU The TSF shall explicitly deny access of subjects to objects based on the following additional *WEBUSER SFP* rules:

a) [selection: [assignment: *rules, based on security attributes, that explicitly deny access of subjects to objects*], "no additional rules"]

Dependencies: FDP_ACC.1 Subset access control
FMT_MSA.3 Static attribute initialization

FDP_RIP.1: Subset residual information protection

Hierarchical to: FDP_RIP.1

FDP_RIP.1.1 The TSF shall ensure that any previous information content of a resource is made unavailable upon the *deallocation of the resource from* the following objects: [

a) *controlled-access content and;*

b) [assignment: *list of other objects*]].

Dependencies: No dependencies

FDP_UCT.1/WU: Basic Data Exchange Confidentiality/WU

Hierarchical to: No other components

FDP_UCT.1.1/WU The TSF shall enforce the *WEBUSER SFP* to be able to *receive* **controlled-access content** in a manner protected from unauthorized disclosure.

Dependencies: [FTP_ITC.1 Inter-TSF trusted channel, or
FTP_TRP.1 Trusted path]
[FDP_ACC.1 Subset access control, or
FDP_IFC.1 Subset information flow control]

FDP_UIT.1/WU: Data Exchange Integrity/WU

Hierarchical to: No other components

FDP_UIT.1.1/WU The TSF shall enforce the *WEBUSER SFP* to be able to *transmit and receive* user data in a manner protected from modification errors.

FDP_UIT.1.1/WU The TSF shall be able to determine on receipt of user data, under the *WEBUSER SFP*, whether modification has occurred.

Dependencies: [FDP_ACC.1 Subset access control, or
FDP_IFC.1 Subset information flow control]
[FTP_ITC.1 Inter-TSF trusted channel, or
FTP_TRP.1 Trusted path]

Application Note:

The intent of the FDP_UCT and FDP_UIT elements in this SFP are to require the use of an encrypting protocol during transmission of content to which access control has been applied (i.e., controlled-access content).

#### 5.1.1.3 FIA: Identification and authentication

FIA_AFL.1-NIAP-0425: Authentication failure handling

Hierarchical to: No other components

FIA_AFL.1.1-NIAP-0425 The TSF shall detect when [assignment: *an administrator configurable integer*] of unsuccessful authentication attempts occur related to *web user login*.

FIA_AFL.1.2-NIAP-0425 When the defined number of unsuccessful authentication attempts has been met or surpassed, the TSF shall prevent the *web users* from performing activities that require authentication until an action is taken by the administrator.

Dependencies: FIA_UAU.1 Timing of authentication

Application Note:

Unsuccessful authentication refers to the presenting of invalid authentication information. This includes invalid passwords as well as invalid certificates.

When a web user attempts to access a URI protected by a realm (controlled-access content), the web server will issue a 401 response (*Not Authorized)* with a WWW-Authenticate header or a 407 response (*Proxy Authentication Required)* with a Proxy-Authenticate header. The browser will provide the user a prompt labeled with the realm and allow the user to enter a username and password (RFC 2617 describes two authorization schemes, basic and digest, but allows others to be defined). Most newer browsers (Amaya, Internet Explorer 5.0, Mozilla 0.9.7 and Opera 6. support basic and digest authorization, however some implementation are not compatible with the standard and thus, are not interoperable. Thus, digest authorization is not normally used.

For basic authorization, the browser will send the server the username, password pair as a base 64 encoded string. While the username and password are not passed in the open, the base 64 encoding offers no protection from a threat agent capturing the message and decoding it. For this reason, to be compliant with this PP, all controlled-access content must be protected using SSL/TLS while in transit.

FIA_ATD.1: User Attribute Definition

Hierarchical to: No other components

FIA_ATD.1.1 The TSF shall maintain the following list of security attributes belonging to **each web user**:

a) *Identification of the user and*

b) *Credentials used to authenticate the user*

Dependencies: No dependencies

Operations Note:

This was refined to clarify the meaning of the phrase "individual users" to be those users within the TSC.

Application Note:

The web server is an application and can not protect the identification of the user or the credentials used to authenticate the user while they are stored (e.g. on a disk drive, cached in a page file etc.) This PP includes FDP requirements to protect this information and the TOE from potential threat agents.

FIA_UAU.1: Timing of Authentication

Hierarchical to: No other components

FIA_UAU.1.1 The TSF shall allow *the GET operation on public content* on behalf of the user to be performed before the user is authenticated.

FIA_UAU.1.2 The TSF shall require each user to be successfully authenticated before allowing any other TSF-mediated actions on behalf of that user.

Dependencies: FIA_UID.1 Timing of identification

FIA_UID.1: Timing of Identification

Hierarchical to: No other components

FIA_UID.1.1 The TSF shall allow *only access of content designated as public* on behalf of the user to be performed before the user is identified.

FIA_UID.1.2 The TSF shall require each user to have been successfully identified before allowing other any TSF-mediated actions on behalf of that user.

Dependencies: No dependencies

Application Note:

If the underlying IT environment provides identification services for content providers and administrators, it is acceptable for FIA_UID.1.2 to be satisfied by the presentation and verification of those credentials.

FIA_USB.1-NIAP-0351: User-Subject Binding

Hierarchical to: No other components

FIA_USB.1.1-NIAP-0351 The TSF shall associate all user security attributes with subjects acting on behalf of that user.

Dependencies: FIA_ATD.1 User attribute definition

#### 5.1.1.4 FMT: Security management

FMT_MOF.1: Management of Security Functions Behavior

Hierarchical to: No other components

FMT_MOF.1.1 The TSF shall restrict the ability to *enable, disable, and modify the behavior of* the *TOE audit functions* to *the administrator*.

Dependencies: FMT_SMF.1 Specification of management functions
FMT_SMR.1 Security roles

Application Note:

In FMT_MOF.1.1 the ability to enable, disable and modify the behavior of the TOE audit functions is restricted to the administrator. In this requirement, the administrator has control over enabling/disabling the TOE's generation of audit records. These audit records are submitted to the host operating system is an unspecified (though evaluated[5]) mechanism (e.g. syslog, Windows Event Manager etc.). The administrator also has

[5] The underlying operating system and its hardware platform must have been previously evaluated against an NSA approved Protection Profile. Furthermore, FIPS 140-2 identifies specific protection profiles.

complete control over the operating system which provides the audit interface to the TOE.

FMT_MSA.1: Management of security attributes

Hierarchical to: No other components

FMT_MSA.1.1 The TSF shall enforce the *WEBUSER SFP* to restrict the ability to [selection: *change_default, query, modify, delete,* [assignment: *other operations*]] the security attributes [assignment: *list of security attributes*] to

1. administrators;
2. [assignment: *other authorized identified roles*].

Dependencies: [FDP_ACC.1 Subset access control or
FDP_IFC.1 Subset information flow control]
FMT_SMF.1 Specification of management functions
FMT_SMR.1 Security roles

FMT_MSA.2: Secure Security Attributes

Hierarchical to: No other components

FMT_MSA.2.1 The TSF shall ensure that only secure values are accepted for security attributes.

Dependencies: ADV_SPM.1 Informal TOE security policy model
[FDP_ACC.1 Subset access control or
FDP_IFC.1 Subset information flow control]
FMT_MSA.1 Management of security attributes
FMT_SMR.1 Security roles

FMT_MSA.3-NIAP-0429: Static attribute initialization

Hierarchical to: No other components

FMT_MSA.3.1-NIAP-0429 The TSF shall enforce the *WEBUSER SFP* to provide *restrictive* default values for security attributes that are used to enforce the SFP.

FMT_MSA.3.2-NIAP-0429 The TSF shall allow the *Web Server Administrator* to specify alternative initial values to override the default values when an object or information is created.

Dependencies: FMT_MSA.1 Management of security attributes
FMT_SMR.1 Security roles

FMT_MTD.1: Management of TSF Data

Hierarchical to: No other components

FMT_MTD.1.1 The TSF shall restrict the ability *to change the default, query, modify, delete, clear, and define the TOE content* to the *Web Server Administrator and Content Providers*.

Dependencies: FMT_SMF.1 Specification of management functions
FMT_SMR.1 Security roles

FMT_REV.1: Revocation

Hierarchical to: No other components.

FMT_REV.1.1 The TSF shall restrict the ability to revoke security attributes associated with *the web users, content providers, and controlled objects* within the TSC to *Web Server Administrator*.

FMT_REV.1.2 The TSF shall enforce the following rules: [assignment: *specification of revocation rules*].

Dependencies: FMT_SMR.1 Security Roles

FMT_SMF.1: Specification of Management Functions

Hierarchical to: No other components.

FMT_SMF.1.1 The TSF shall be capable of performing the following security management functions:

a) managing of the authentication data by an administrator;

b) managing of the authentication data by the associated user;

c) managing the list of actions that can be taken before the user is authenticated;

d) managing realms by an administrator;

e) managing content providers by an administrator;

f) assigning content providers to realms by an administrator

g) assigning web users to realms

h) managing changes to cryptographic key attributes;

i) managing the conditions under which abstract machine test occurs;

j) managing the conditions under which TSF self testing occurs;

k) specification of the time of user inactivity after which termination of the interactive session occurs for an individual user;

l) specification of the default time of user inactivity after which termination of the interactive session occurs.

m) [assignment: *list of security management functions to be provided by the TSF*].

Dependencies: No Dependencies

FMT_SMR.1: Security Roles

Hierarchical to: No other components.

FMT_SMR.1.1 The TSF shall maintain the following roles:

a) Web Server Administrator

b) Content Provider

c) Web User

FMT_SMR.1.2 The TSF shall be able to associate users with roles.

Dependencies: FIA_UID.1 Timing of identification

### 5.1.2 Cryptomodule Security Functional Requirements

**Table 5-3: Cryptomodule Security Functional Requirements**

| Identifier | Description |
|---|---|
| FCS: Cryptographic Support | |
| FCS_BCM_EXP.1 | Baseline Cryptographic Module |
| FCS_CBP_EXP.1 | Cryptographic Bypass |
| FCS_CKM.1 | Cryptographic Key Generation (using Random Number Generator) |
| FCS_CKM.4 | Cryptographic Key Destruction |
| FCS_CKM_EXP.1 | Cryptographic Key Establishment |
| FCS_CKM_EXP.2 | Discrete Logarithm Key Agreement |
| FCS_CKM_EXP.3 | Elliptic Curve Key Agreement |
| FCS_CKM_EXP.4 | Key Transport |
| FCS_CKM_EXP.5 | Manual Loading of Key |
| FCS_CKM_EXP.6 | Automated Loading of Key |
| FCS_COP.1(1) | Cryptographic operation (Encryption) |
| FCS_COP.1(2) | Cryptographic operation (Digital Signature) |

| Identifier | Description |
|---|---|
| FCS_COP.1(3) | Cryptographic operation (Hashing) |
| FCS_COP.1(4) | Cryptographic operation (Random Number Generation) |
| FCS_KXP_EXP.1 | Export of Keying Material |
| FPT: Protection of the TOE Security Functions | |
| FPT_RVM.1 | Non-bypassability of the TSP |
| FPT_SEP_EXP.1 | Application Domain Separation |
| FPT_TST.1/CR | TSF Testing (Cryptography and Critical Functions) |
| FPT_TST_EXP.1/KG | TSF Testing (Key Generation Components) |
| FTA: TOE Access | |
| FTA_SSL.3 | TSF-initiated session termination |
| FTP: Trusted Path | |
| FTP_ITC.1 | Inter-TSF trusted channel |

#### 5.1.2.1 FCS: Cryptographic Support

The cryptographic requirements are structured to accommodate use of FIPS 140-2-validated cryptographic modules (also called cryptomodules) in meeting the requirements. Since the FIPS 140-2 scheme does not cover all aspects of all algorithms, a convention is needed to distinguish the cryptographic functionality that the TSF is required to provide that cannot be provided by a FIPS-validated cryptomodule from cryptographic functionality that can be provided via a FIPS-validated cryptomodule. In the following text and requirements, "cryptomodule" is used in the very specific sense that it is

a module that is FIPS 140-2 validated (to comply with FCS_BCM_EXP below);

a module implementing validated NIST-approved security functions; and

a module containing cryptographic functionality available in a NIST-approved mode.

It is the intent of these requirements (and the requirements are worded such) that whenever cryptographic functionality that can be FIPS-validated is required, that functionality be implemented in a cryptomodule. This means that when key management requirements (including key generation) are present, the key management functionality must be present in the cryptomodule. As an example, cryptomodules implementing AES must generate their own key.

It is important to note to vendors and end users that any IT entity that is used to protect National Security Information, and employs cryptography as a protection mechanism, will require the TOE's key management techniques to be approved by NSA when the TOE is fielded.

FCS_BCM_EXP.1: Baseline Cryptographic Module

Hierarchical to: No other components.

FCS_BCM_EXP.1.1 The cryptomodule shall perform all

a) data encryption and decryption,

b) digital signature generation and verification,

c) cryptographic hashing and

d) random number generation functions

used by the TSF

FCS_BCM_EXP.1.2 The cryptomodule shall perform the specified cryptographic functions in a NIST-approved mode of operation.

FCS_BCM_EXP.1.3 The cryptomodule module shall be FIPS PUB 140-2 validated.

FCS_BCM_EXP.1.4 The cryptomodule implementation shall have a minimum overall rating of FIPS PUB 140-2 Security Level 1.

FCS_CBP_EXP.1: Cryptographic Bypass

Hierarchical to: No other components.

FCS_CBP_EXP.1 If a **cryptomodule** module implements a cryptographic bypass capability (where services are provided without cryptographic processing e.g., transferring plaintext through the module without encryption), then

a) two independent internal actions shall be required to activate the capability to prevent the inadvertent bypass of plaintext data due to a single error (e.g., two different software or hardware flags are set, one of which may be user-initiated), and

b) the module shall show status to indicate whether

1) the bypass capability is not activated, and the module is exclusively providing services with cryptographic processing (e.g., plaintext data is encrypted),

2) the bypass capability is activated and the module is exclusively providing services without cryptographic processing (e.g., plaintext data is not encrypted), or

3) the bypass capability is alternately activated and deactivated and the module is providing some services with cryptographic processing and some services without cryptographic processing (e.g., for modules with multiple communication channels, plaintext data is or is not encrypted depending on each channel configuration).

FCS_CKM.1: Cryptographic Key Generation (using Random Number Generator)

FCS_CKM.1.1 The cryptomodule shall generate symmetric cryptographic keys using a NIST-approved Random Number Generator for all key sizes that meet one of the standards defined in Annex C to FIPS 140-2.

Application Note:

Annex C to FIPS 140-2 defines NIST-approved random number generation algorithms. Each of the algorithms is defined in an associated standard listed in the Annex.

### FCS_CKM.4: Cryptographic Key Destruction

FCS_CKM.4.1 The **cryptomodule** shall destroy cryptographic keys in accordance with a cryptographic key zeroization method that meets the following:

a) Key Zeroization Requirements in FIPS PUB 140-2 Key Management Security Level 1;

b) Zeroization of all plaintext cryptographic keys and all other critical cryptographic security parameters shall be immediate and complete;

c) For embedded cryptographic modules, the zeroization shall be executed by overwriting the key/critical cryptographic security parameter storage area three or more times with an alternating pattern;

d) If the cryptographic module contains any doors or removable covers or if a maintenance access interface is defined, then the module shall contain tamper response and zeroization circuitry. The tamper response and zeroization circuitry shall immediately zeroize all plaintext secret and private keys and critical cryptographic security parameters when a door is opened, a cover is removed, or when the maintenance access interface is accessed. The tamper response and zeroization circuitry shall remain operational when plaintext secret and private cryptographic keys or CSPs are contained within the cryptographic module; and

e) When transferring any key/CSP to another location, the TSF shall overwrite each intermediate storage area for private cryptographic keys, plaintext cryptographic keys, and all other critical security parameters three or more times with an alternating pattern.

Application note:

The item e applies to locations that are used when the keys/parameters are copied during processing, and not to the locations that are used for storage of the keys, which are specified in items c and d. The temporary locations could include memory registers, physical memory locations, and even page files and memory dumps.

FCS_CKM_EXP.1: Cryptographic Key Establishment

FCS_CKM_EXP.1.1 The **cryptomodule** shall provide [selection: *Discrete Logarithm Key Agreement, Elliptic Curve Key Agreement, Key Transport, Manual Loading*] key establishment technique(s) in accordance with [assignment: *the following FIPS approved key establishment techniques applicable to FIPS 140-2.*].

Application Note:

FIPS PUB 140-2 Annex D provides a list of the FIPS Approved key establishment techniques applicable to FIPS PUB 140-2. The ST author must select one or more key establishment techniques.

FCS_CKM_EXP.2: Discrete Logarithm Key Agreement

Hierarchical to: No other components.

FCS_ CKM_EXP.2.1 If the **cryptomodule** supports the Discrete Logarithm Key Agreement key establishment technique then

a) TSF shall provide the capability to act as the initiator or responder (that is, act as Party U or Party V as defined in the standard) to agree on cryptographic keys of all sizes using the [selection: dhStatic, dhEphem, dhOneFlow, dhHybrid1, dhHybrid2, dhHybridOneFlow, MQV1, MQV2] key agreement scheme where domain parameter p is a prime of [assignment: length of prime “p” in bits (1024 or greater)] bits and domain parameter q is a prime of [assignment: length of prime “q” in bits (160 or greater)], and that conforms with ANSI X9.42-2001, Public Key Cryptography for the Financial Services Industry: Agreement of Symmetric Keys Using Discrete Logarithm Cryptography.

b) The **cryptomodule** shall conform to ANSI X9.42-2001 using a NIST-approved Message Authentication Code (MAC) function, a NIST-approved Random Number generation function, and a NIST-approved Hashing function.

c) The choices and options used in conforming to the key agreement scheme(s) are as follows: [assignment: *options that the TSF implements when implementing the selected key agreement schemes, including options for any prerequisite or dependant functions (e.g., domain parameter generation and validation)*].

Application Note:

It should be noted that the actual key size of the symmetric key agreed to when using this scheme will be a function of the algorithm that will be using the key, as specified in FCS_COP.1 (1).

In the selection in paragraph a), one or more of the schemes should be chosen by the ST writer, based on what schemes the TOE implements. Note that the requirement is for the cryptomodule to be able to act as either party (as detailed in the standard) for the chosen scheme(s).

The two assignments are used to specify the number of bits used for the domain parameters p and q (which are primes). The requirement above indicates that p must be a prime of at least 1024 bits, while q must be a prime of at least 160 bits. The ST writer should fill in the appropriate number of bits based on the implementation. This applies if the implementation generates its own domain parameters, or if it obtains the domain parameters in some other way (e.g., hard-coded, obtained from an outside authority).

In the X9.42-2001 standard there are several sections that are marked “optional”, or where a choice is given. Choices are, for example, how the domain parameters are obtained (generated or obtained from some other entity). Another example is the key derivation function that is implemented. ST writers should use the assignment to provide sufficient information so that 1) it is possible to test the implementation of the function in a repeatable fashion, and 2) readers (consumers) of the ST understand exactly what is done by the key agreement schemes implemented. The ST author should ensure that all of the prerequisite options/choices, as well as choices/options in dependant functions, are covered in the assignment.

FCS_CKM_EXP.3: Elliptic Curve Key Agreement

Hierarchical to: No other components.

FCS_CKM_EXP.3.1 If the **cryptomodule** supports Elliptic Curve Key Agreement then

a) The TSF shall provide the capability to act as the initiator or responder (that is, act as Party U or Party V as defined in the standard) to agree on cryptographic keys of all sizes using the [selection: *Ephemeral Unified Model, 1-Pass Diffie-Hellman, Static Unified Model, Combined Unified Model with Key*

*Confirmation, 1-Pass Unified Model, Full Unified Model, Full Unified Model with Key Confirmation, Station-to-Station, 1-Pass MQV, Full MQV, Full MQV with Key Confirmation*] key agreement scheme using Elliptic Curves with the order of the base point being a [assignment: *length of the order of the base point "n" in bits (160 or greater)*]-bit value, and conforms to ANSI X9.63-2001, Public Key Cryptography for the Financial Services Industry: Key Agreement and Key Transport Elliptic Curve Cryptography.

b) The TSF shall conform to the standard using a NIST-approved MAC function, a NIST-approved Random Number generation function, and a NIST-approved Hashing function.

c) The choices and options used in conforming to the key transport scheme(s) are as follows: [assignment: *options that the TSF implements when implementing the selected key transport schemes, including options for any prerequisite or dependant functions (e.g., domain parameter generation and validation*].

Application Note:

This element of the top-level selection applies to automated key agreement schemes where an exchange occurs between the TOE and another IT entity that results in both entities having the same secret key without ever having passed that key between the two entities. This is in contrast to key transport schemes, where key is actually passed between two IT entities. This is also distinct from key loading, where the user is either directly inputting or receiving key, or an automated device (token, PC card, etc.) is inputting or receiving key.

It should be noted that the actual key size of the symmetric key agreed to when using this scheme will be a function of the algorithm that will be using the key, as specified in FCS_COP.1 (1).

In the selection in paragraph a), one or more of the schemes should be chosen by the ST writer, based on what schemes the TOE implements. Note that the requirement is for the TSF to be able to act as either party (as detailed in the standard) for the chosen scheme(s) where the schemes are asymmetric.

The assignment is used to specify the number of bits used for the domain parameter n, which is the order of the base point of the curve chosen (the standard uses "n" to denote this value). The requirement above indicates that n must be at least a 160-bit value. The ST writer should fill in the appropriate number of bits based on the implementation. This applies if the implementation generates its own domain parameters, or if it obtains the domain parameters in some other way (e.g., hard-coded, obtained from an outside authority).

Application Note: In the X9.63-2001 standard there are several sections that are marked "optional", or where a choice is given. Choices are, for example, in the domain parameter generation and validation section (Section 5.1) where domain parameters can be generated over Fp or over F2m. Another example is the Diffie-Hellman primitive (Standard or Modified) that is implemented. ST writers should use the assignment to provide sufficient information so that 1) it is possible to test the implementation of the function in a repeatable fashion, and 2) readers (consumers) of the ST understand exactly what is done by the key agreement schemes implemented. The ST author should ensure that all of the prerequisite options/choices, as well as choices/options in dependant functions, are covered in the assignment.

FCS_CKM_EXP.4: Key Transport

Hierarchical to: No other components.

FCS_CKM_EXP.4.1 If the **cryptomodule** supports Key Transport then

a) The TSF shall provide (act as the initiator) and accept (act as the responder) cryptographic keys to/from another IT Entity using the [selection: *1-Pass Transport Scheme; 3-Pass Transport Scheme; both the 1-Pass and 3-Pass Transport Schemes*] using Elliptic Curves with the order of the base point being a [assignment: *lenght of modulus "n" in of bits (160 or greater)*]-bit value in a manner that conforms with ANSI X9.63-2001, Public Key Cryptography for the Financial Services Industry: Key Agreement and Key Transport Elliptic Curve Cryptography.

b) The **cryptomodule** shall conform to the standard using a NIST-approved MAC function, a NIST-approved Random Number generation function, and a NIST-approved Hashing function.

c) The choices and options used in conforming to the key transport scheme(s) are as follows: [assignment: *options that the TSF implements when implementing the selected key transport schemes, including options for any prerequisite or dependant functions (e.g., domain parameter generation and validation*].

Application Note:

In the selection in paragraph a), one or more of the schemes should be chosen by the ST writer, based on what schemes the TOE implements. Note that the requirement is for the TSF to be able to act as either party (as detailed in the standard) for the chosen scheme(s).

The assignment is used to specify the number of bits used for the domain parameter n, which is the order of the base point of the curve chosen (the standard uses "n" to denote this value). The requirement above indicates that n must be at least a 160-bit value. The ST writer should fill in the appropriate number of bits based on the implementation. This applies if the implementation generates its own domain parameters, or if it obtains the domain parameters in some other way (e.g., hard-coded, obtained from an outside authority).

In the X9.63-2001 standard there are several sections that are marked "optional", or where a choice is given. Choices are, for example, in the domain parameter generation and validation section (Section 5.1) where domain parameters can be generated over Fp or over F2m. Another example is the Diffie-Hellman primitive (Standard or Modified) that is implemented. ST writers should use the assignment to provide sufficient information so that 1) it is possible to test the implementation of the function in a repeatable fashion, and 2) readers (consumers) of the ST understand exactly what is done by the key agreement schemes implemented. The ST author should ensure that all of the prerequisite options/choices, as well as choices/options in dependant functions, are covered in the assignment.

FCS_CKM_EXP.5: Manual Loading of Key

Hierarchical to: No other components.

FCS_CKM_EXP.5.1 If the **cryptomodule** supports the manual loading of keying material then the cryptomodule shall be able to accept as input cryptographic keys in accordance with [assignment: *a specified manual cryptographic key distribution method*] using [assignment: *FIPS-approved Key Management techniques*] that meet the FIPS 140-2 Key Management Security Levels 1, Key Entry and Output.

Application Note:

This requirement applies to the case where a human is either typing key into the cryptomodule, or the cryptomodule is outputting key to a display, for instance. The distinguishing feature is that the transaction is between a human and the cryptomodule, and not between the cryptomodule and another IT device or IT media.

The manual entry of keying material into the cryptomodule must be in accordance with FIPS PUB 140-2 Key Management Security Level 1.

FCS_CKM_EXP.6: Automated Loading of Key

Hierarchical to: No other components.

FCS_ECK_EXP.6.1 If the **cryptomodule** supports Automated Loading Key then

a) The cryptomodule shall [selection:
   1) *be able to accept as input*;
   2) be able to output in the following circumstances [assignment: circumstances under which the cryptomodule will output a key]] cryptographic keys in accordance with a specified electronic cryptographic key distribution method using NIST-approved Key Management techniques;

b) The electronic device is directly attached by [selection: internal bus, serial port, USB port, audio device, assignment: [other non-network physical device]] to the TSF;

c) The TSF shall perform key error detection scheme on keys input via electronic methods using [selection: parity check, [assignment: other key error detection scheme]; and

d) FIPS 140-2 Key Management Security Levels 1, Key Entry and Output.]

Application Note:

This element of the top-level selection applies to automated key loading device. In the case where key is being transferred from the device to the TSF the key is being “input”. In the case where the key is being transferred from the TSF to the device (for instance, a CA loading a user’s private key into a token device) the key is being “output.”

The selection should be used by the ST author to indicate whether the cryptomodule is capable of accepting key, capable of outputting key, or both. In the case where the key is output, the ST author should use the assignment to detail the conditions under which key is output from the cryptomodule (for example, only during a certain type of key generation activity).

An example of a device attached by an internal bus would be a floppy device used for keys transported on floppy disks.

Application Note:

The ST writer should indicate what error detection scheme is employed. The requirement above refers to errors in parity or structure of the key; it does not necessarily require checks on key “goodness”, length, format, etc.

Note that this requirement mandates that cryptomodules in the TSF have the ability to perform automated key input/output, and that this capability has been through the FIPS validation process.

The ST author selects one or more of the identified methods (i.e., the two key agreement schemes, key transport, manual loading or automated loading) used to establish cryptographic keys in the TOE.

FCS_COP.1(1): Cryptographic operation (Encryption)

Hierarchical to: No other components.

FCS_COP.1.1(1) The **cryptomodule** shall perform *data encryption and decryption services* in accordance with [selection: *Triple Data Encryption Algorithm (TDEA), Advanced Encryption Standard (AES)*] and cryptographic key sizes *of 128 bits or greater* that meet the following: [selection:

a) *FIPS 46-3, Data Encryption Standard (DES)* –and– *ANSI X9.52-1998, Triple Data Encryption Algorithm Modes of Operation* –or–

b) *FIPS 197, Advanced Encryption Standard (AES)*,

c) *SP 800-38A, Recommendation for Block Cipher Modes of Operation*].

| | | |
|---|---|---|
| Dependencies: | [FDP_ITC.1<br>or | Import of user data without security attributes |
| | FCS_CKM.1 | Cryptographic key generation] |
| | FCS_CKM.4 | Cryptographic key destruction |
| | FMT_MSA.2 | Secure security attributes |

Application Note:

The ST author should specify the modes in which the cryptomodule operates in the TOE. Note that these modes must be available in the NIST-approved operation mode of the cryptomodule. SP 800-38A ("Recommendation for Block Cipher Modes of Operation") specifies five confidentiality modes that are used with any approved block cipher. The modes in SP 800-38A are updated versions of the ECB, CBC, CFB, and OFB modes that are specified in FIPS Pub. 81; in addition, SP 800-38A specifies the CTR mode.

FCS_COP.1(2): Cryptographic operation (Digital Signature)

Hierarchical to: No other components.

FCS_COP.1.1(2) The **cryptomodule** shall perform *digital signature generation and verification* in accordance with [selection:

a) *Digital Signature Algorithm (DSA) with a key size (modulus) of 1024 bits or greater;*

b) *RSA Digital Signature Algorithm (rDSA with odd e) with a key size (modulus) of 1024 bits or greater;* or

c) *Elliptic Curve Digital Signature Algorithm (ECDSA) with a key size of 160 bits or greater*]

that meet the following: [selection:

a) *FIPS PUB 186-2, Digital Signature Standard, for signature creation and verification processing; and ANSI Standard X9.42-2001, Public Key Cryptography for the Financial Services Industry: Agreement of Symmetric Keys Using Discrete Logarithm Cryptography for generation of the domain parameters;*

b) *ANSI X 9.31-1998 (May 1998), Digital Signatures Using Reversible Public Key Cryptography for the Financial Services Industry (rDSA);* or

c) *ANSI X9.62-1-1998, Public Key Cryptography for the Financial Services Industry: Elliptic Curve Digital Signature Algorithm (ECDSA)*].

| | | |
|---|---|---|
| Dependencies: | [FDP_ITC.1 | Import of user data without security attributes |
| | or | |
| | FCS_CKM.1 | Cryptographic key generation] |
| | FCS_CKM.4 | Cryptographic key destruction |
| | FMT_MSA.2 | Secure security attributes |

Application Note:

In the X9.31-1998 standard there are several sections that are marked “optional”, or where a choice is given. For instance, the public verification exponent “e” can be fixed or randomly generated. Another instance is that the procedure in section 4.1.2.1 can be followed to generate the primes p and q, or another procedure followed as long as the primes generated meet the conditions in section 4.1.2. The goal of the assignment is to provide sufficient information such that 1) it is possible to test the implementation of the function in a repeatable fashion, and 2) readers (consumers) of the ST understand exactly what is done by the rDSA implementation. The ST author should ensure that all of the prerequisite options/choices, as well as choices/options in dependant functions, are covered in the assignment.

The Elliptic Curve Digital Signature Algorithm requirement above indicates the number of bits used for the domain parameter n, which is the order of the base point of the curve chosen (the standard uses “n” to denote this value). That n must be at least a 160-bit value. The ST writer should fill in the appropriate number of bits based on the implementation. This applies if the implementation generates its own domain parameters, or if it obtains the domain parameters in some other way (e.g., hard-coded, obtained from an outside authority).

FCS_COP.1(3): Cryptographic operation (Hashing)

Hierarchical to: No other components.

FCS_COP.1.1(3) The **cryptomodule** shall perform *cryptographic hashing functions* in accordance with [assignment: ***NIST approved*** *cryptographic* ***hashing*** *algorithm(s)*] that meet the following: [assignment: *list of* ***NIST approved cryptographic hashing*** *standards*].

| | | |
|---|---|---|
| Dependencies: | [FDP_ITC.1 | Import of user data without security attributes |
| | or | |
| | FCS_CKM.1 | Cryptographic key generation] |
| | FCS_CKM.4 | Cryptographic key destruction |
| | FMT_MSA.2 | Secure security attributes |

Application Note:

Whenever a referenced standard calls for a cryptographic hashing capability (e.g., SHA-1), this requirement specifies the subset of cryptographic hashing functions (those that are FIPS-validated) that are acceptable. Note that the hashing function does not have to be implemented in the cryptomodule that is performing the cryptographic operation. Also note that this requirement is not calling for the hashing functionality to be made generally available (e.g., to untrusted users via an API).

FCS_COP.1(4): Cryptographic operation (Random Number Generation)

Hierarchical to: No other components.

FCS_COP.1.1(4) The **cryptomodule** shall perform *random number generation* in accordance with [assignment: ***NIST approved random number generation*** *algorithm(s)*] that meet the following: [assignment: *list of* ***NIST approved random number generation*** *standards*].

| | | |
|---|---|---|
| Dependencies: | [FDP_ITC.1 | Import of user data without security attributes |
| | or | |
| | FCS_CKM.1 | Cryptographic key generation] |
| | FCS_CKM.4 | Cryptographic key destruction |
| | FMT_MSA.2 | Secure security attributes |

Application Note:

Whenever a referenced standard calls for a random number generation capability, this requirement specifies the subset of random number generators (those that are FIPS-validated) that are acceptable. Note that the random number generator does not have to be implemented in the cryptomodule that is performing the cryptographic operation. The random number generator does not have to be made generally available (e.g., to untrusted users via an API).

FCS_KXP_EXP.1: Export of Keying Material

Hierarchical to: No other components.

FCS_FMK_EXP.1.1 The **cryptomodule** shall export keying material in the following circumstances [assignment: *circumstances under which the cryptomodule will output a key*] in accordance with [assignment: *a specified manual cryptographic key distribution method*] using [assignment: *NIST-approved Key Management techniques*] that meets the FIPS 140-2 Key Management Security Levels 1, Key Entry and Output.

Dependencies: TBD

### 5.1.2.2 FPT: Protection of the TOE Security Functions

FPT_RVM.1: Non-bypassability of the TSP

Hierarchical to: No other components

FPT_RVM.1.1 The **cryptographic module** shall ensure that cryptographic security policy enforcement functions are invoked and succeed before each function within the **cryptographic module** scope of control is allowed to proceed.

Dependencies: No dependencies

FPT_SEP_EXP.1: Application Domain Separation

Hierarchical to: No other components.

FPT_SEP_EXP.1.1 The TSF shall maintain a security domain that protects it from interference and tampering by untrusted subjects initiating actions through its own TSFI.

FPT_SEP_EXP.1.2 The TSF shall enforce separation between the security domains of subjects in the TOE Scope of Control.

Dependencies: No dependencies

FPT_TST.1/CR: TSF Testing (Cryptography and Critical Functions)

Hierarchical to: No other components

FPT_TST.1.1/CR The TSF shall run a suite of self-tests, at the following times, in accordance with FIPS PUB 140-2, Level 1 (as identified in Table 5 1) to demonstrate the correct operation of the indicated functions of the TOE.

a) Testing Times: during initial start-up (on power on); at the request of the administrator (on demand); under the following conditions [assignment: other conditions under which the cryptographic self tests shall be run]; and periodically.

b) Functions to be tested: cryptographically software/firmware; cryptographic algorithms; RNG/PRNG; other FIPS PUB 140-2 critical functions; and [assignment: list of all critical security functions].

**Table 5-4: Interpretation of FIPS PUB 140-2 Self Tests**

| Self-Tests | FIPS-140 Security Level 1 |
|---|---|
| Software/Firmware Integrity Tests | on power on<br>on demand<br>conditional |
| Cryptographic Algorithm Tests | on power on<br>on demand<br>conditional |
| Other FIPS PUB 140-2 critical functions tests and other tests as determined by FIPS PUB 140-2, Appendix A | on power on<br>on demand<br>conditional |
| Statistical RNG/PRNG tests | on power on<br>on demand |

FPT_TST.1.2/CR The TSF shall provide the administrators with the capability to verify the integrity of cryptographically related TSF data.

FPT_TST.1.3/CR The TSF shall provide the administrators with the capability to verify the integrity of stored cryptographically related TSF executable code.

Dependencies: FPT_AMT.1 Abstract machine testing

Application Note:

The ST author fills in the conditions under which the self-tests are run by consulting FIPS 140-2 as well as to reflect capabilities of the TOE.

FPT_TST_EXP.1/KG: TSF Testing (Key Generation Components)

Hierarchical to: No other components

FPT_TST_EXP.1.1/KG The TSF shall run a suite of self-tests immediately after generation of a key to demonstrate correct operation of each key generation component. If any of these tests fails, that generated key shall not be used, the cryptographic module shall react as required by FIPS PUB 140-2 for failing a self-test, and this event will be audited.

FPT_TST_EXP.1.2/KG The TSF shall provide the Administrator with the capability to verify the integrity of TSF data related to the key generation.

FPT_TST_EXP.1.3/KG The TSF shall provide the Administrator with the capability to verify the integrity of stored TSF executable code related to the key generation.

Dependencies: FPT_AMT.1 Abstract machine testing

Application Note:

Key generation components are those critical elements that compose the entire key generation process (e.g., any algorithms, any RNG/PRNGs, any key generation seeding processes, etc.).

#### 5.1.2.3 FTA: TOE Access

FTA_SSL.3: TSF-initiated session termination

Hierarchical to: No other components

FTA_SSL.3.1 The TSF shall terminate an interactive **HTTPS** session after a [*Web Server Administrator-configurable time interval of session inactivity*].

Dependencies: No dependencies

Application Note:

HTTP and HTTPS are state-less protocols that were designed to allow an anonymous user to request a document and the server to service that request. Several mechanisms (e.g. session cookies, URL rewriting, SSL ID tracking, etc.) have been devised to bind a set of requests to a user or browser, providing HTTP sessions. To meet FTA_SSA_EXP.1.1, the TOE must be able to detect a period of inactivity in each HTTP session and terminate any session if that period exceeds a Web Server Administrator determined period of time.

#### 5.1.2.4 FTP: Trusted Path

FTP_ITC.1: Inter-TSF trusted channel

Hierarchical to: TBD

FTP_ITC.1.1 The TSF shall provide a communication channel between itself and a remote trusted IT product that is logically distinct from other communication channels and provides assured identification of its end points and protection of the channel data from modification or disclosure.

FTP_ITC.1.2 The TSF shall permit either the TSF or the remote trusted IT product to initiate communication via the trusted channel.

FTP_ITC.1.3 The TSF shall initiate communication via the trusted channel for the transmission of controlled-access content.

Dependencies: TBD

## 5.2 Environmental Security Functional Requirements

The notional model is that the Web Server is a software application built on top of an underlying IT platform. This IT platform provides basic controlled access services such as identification and authentication, discretionary access control, residual information protection, protection for the TOE, and a basic level of robustness. Instead of duplicating an already existing profile in this document, the approach taken is to require that the underlying platform be compliant with an appropriate profile. Note that it is acceptable for the TOE to satisfy IT environment requirements; this would be captured in the ST.

**Table 5-5: Environmental Security Functional Requirements**

| Identifier | Name |
|---|---|
| FAU_SAR.1 | Audit Review |
| FAU_SAR.2 | Restricted Audit Review |
| FAU_SAR.3 | Selectable Audit Review |
| FAU_SEL.1-NIAP-0407 | Selective Audit |
| FAU_STG.1-NIAP-0429 | Protected audit trail storage |
| FAU_STG.3 | Action in case of possible audit data loss |
| FAU_STG.NIAP-0414-1-NIAP-0429 | Site-configurable Prevention of audit data loss |
| FDP_ACC.2/CP | Complete Object Access Control/CP |
| FDP_ACF.1-NIAP-0407/CP | Security Attribute Based Access Control/CP |
| FIT_PPC_EXP | IT Environment PP Compliance |
| FPT_SEP_ENV.1 | Domain separation |
| FPT_STM.1 | Reliable time stamps |
| FTA_SSL.1 | TSF-initiated session locking |
| FTA_SSL.2 | User-initiated locking |
| FTA_SSL.3/IN | TSF-initiated termination |
| FTA_TAB.1 | Default TOE Access Banners |

### 5.2.1 FAU: Security Audit

FAU_SAR.1: Audit Review

Hierarchical to: No other components.

FAU_SAR.1.1 The **environment** shall provide the TOE administrators with the capability to read all information contained within the audit record from the audit records.

FAU_SAR.1.1 The **environment** shall provide the audit records in a manner suitable for the user to interpret the information.

Dependencies: TBD

FAU_SAR.2: Restricted Audit Review

Hierarchical to: No other components.

FAU_SAR.2.1 The **environment** shall prohibit all users read access to the audit records, except those users that have been granted explicit read-access.

Dependencies: TBD

FAU_SAR.3: Selectable Audit Review

Hierarchical to: No other components.

FAU_SAR.3.1 The **environment** shall provide the ability to perform searches of audit data based on any of the following:

a) user identity;

b) source subject identity;

c) destination subject identity;

d) ranges of one or more: dates, times, user identities, subject service identifiers, or transport layer protocol;

e) TOE network interfaces; and

f) [selection: [assignment: *other criteria determined by the ST Author*], "*no additional criteria*"].

Dependencies: TBD

Application Note:

It is implied that the Audit Administrator is the only user who can perform these functions, since they are the only users with read access to all of the audit records in the audit trail. Audit data should be capable of being searched and sorted on all criteria specified in a–f, if applicable (i.e., not all criteria will exist in all audit records).

Sorting means to arrange the audit records such that they are "grouped" together for administrative review. For example the Audit Administrator may want all the audit records for a specified source subject identity or range of source subject identities (e.g., IP source address or range of IP source addresses) presented together to facilitate their audit review. If no additional criteria are provided by the TOE to perform searches or sorting of audit data, the ST author selects "no additional criteria".

Operations Note:

This was refined to fix the grammatical introduction to the list.

FAU_SEL.1-NIAP-0407: Selective Audit

Hierarchical to: No other components.

FAU_SEL.1.1-NIAP-0407 The **environment** shall be able to include or exclude auditable events from the set of audited events based on the following attributes:

a) user identity;

b) network identifier;

c) subject service identifier;

d) event type;

e) success of auditable security events;

f) failure of auditable security events; and

g) [selection: [assignment: list of additional criteria that audit selectivity is based upon], "no additional criteria"].

Dependencies: TBD

FAU_STG.1-NIAP-0429: Protected audit trail storage

Hierarchical to: No other components

FAU_STG.1.1-NIAP-0429 The **environment** shall protect the stored audit records in the audit trail from unauthorized deletion.

FAU_STG.1.2-NIAP-0429 The **environment** shall be able to prevent modifications to the audit records in the audit trail.

FAU_STG.NIAP-0414-1-NIAP-0429: Site-configurable prevention of audit data loss

Hierarchical to: FAU_STG.4

FAU_STG.NIAP-0414-1-NIAP-0429-1.1) The **environment** shall provide an authorized administrator with the capability to select one or more of the following actions to be taken if the audit trail is full: (

a) prevent auditable events, except those taken by the authorized user with special rights

b) overwrite the oldest stored audit records

c) [selection: [assignment: other actions to be taken in case of audit storage failure], "no additional options"]

FAU_STG.NIAP-0414-1-NIAP-0429.1.2 The **environment** shall overwrite the oldest stored audit records if the audit trail is full and no other action has been selected.

Dependencies: FAU_STG.1 Protected Audit Trail Storage
FMT_MTD.1 Management of TSF Data

Operations Note:

This was refined to make the embedded multiple-choice selection into a list.

Application Note:

The TOE provides the administrator the option of preventing audit data loss by preventing auditable events from occurring. The administrator's actions under these circumstances are not required to be audited. The TOE also provides the administrator the option of overwriting "old" audit records rather than preventing auditable events, which may protect against a denial-of-service attack.

The ST writer should fill in other technology-specific actions that can be taken for audit storage failure (in addition to the two already specified), or select "no additional options" if there are no such technology-specific actions.

FAU_STG.3: Action in case of possible audit data loss

Hierarchical to: No other components

FAU_STG.3.1 The **environment** shall immediately alert the administrators by displaying a message at the local console, [selection: [assignment: other actions determined by the ST author], "none"] if the audit trail exceeds an Administrator-settable percentage of storage capacity.

Dependencies: FAU_STG.1 Protected audit trail storage

Application Note:

The ST Author should determine if there are other actions that should be taken when the audit trail setting is exceeded, and put these in the assignment. If there are no other actions, then the ST Author should select "none".

### 5.2.2 FDP: User Data Protection

FDP_ACC.2/CP: Complete Access Control/CP

Hierarchical to: FDP_ACC.1

FDP_ACC.2.1/CP The **environment** shall enforce the CONTENT-PROVIDER SFP on the following subjects and objects, and upon all operations among subjects and objects covered by this Security Function Policy (SFP):

a) Subjects: Content Providers

b) Objects: Content

FDP_ACC.2.2/CP The **environment** shall ensure that all operations between any subject in the CONTENT-PROVIDER TSC and any object within the CONTENT-PROVIDER TSC are covered by the CONTENT-PROVIDER SFP.

Dependencies: FDP_ACF.1 Security attribute based access control

FDP_ACF.1-NIAP-0407/CP: Security Attribute Based Access Control/CP

Hierarchical to: No other components

FDP_ACF.1.1-NIAP-0407/CP The **environment** shall enforce the CONTENT-PROVIDER SFP to objects based on the identity and group membership of the content provider, the protections on the underlying objects used to create or modify content by the host platform, and the server administrative configuration.

FDP_ACF.1.2-NIAP-0407/CP The **environment** shall enforce the following CONTENT-PROVIDER SFP rules to determine if an operation among controlled subjects and controlled objects is allowed:

a) The **environment** shall restrict the ability to create or modify content to only those content providers authorized by a server administrator.

b) The **environment** shall be capable of limiting the ability to create or modify server executable content to a subset of the authorized content providers.

FDP_ACF.1.3-NIAP-0407/CP The **environment** shall explicitly authorize access of subjects to objects based on the following additional CONTENT-PROVIDER SFP rules:

a) [selection: [assignment: *rules, based on security attributes, that explicitly authorize access of subjects to objects*], "no additional rules"]

FDP_ACF.1.4-NIAP-0407/CP The **environment** shall explicitly deny access of subjects to objects based on the following additional CONTENT-PROVIDER SFP rules:

a) [selection: [assignment: *rules, based on security attributes, that explicitly deny access of subjects to objects*], "no additional rules"]

Dependencies: FDP_ACC.1 Subset access control
FMT_MSA.3 Static attribute initialization

### 5.2.3 FPT: Protection of the TSF

FPT_SEP_ENV.1: Domain separation

FPT_SEP_ENV.1.1 The **environment** shall isolate applications such that any interaction between applications is mediated by a trusted entity.

FPT_SEP_ENV.1.2 The **environment** shall prevent untrusted applications from interfering with the enforcement of the security policy by the trusted entity.

Application Note:

The environment must provide a way to prevent concurrent processes from interfering with each other. Typically, this isolation is provided by the operating system in conjunction with hardware support for multiple CPU privilege levels and memory management.

FPT_STM.1 Reliable time stamps

FPT_STM.1.1 The **environment** shall provide reliable time stamps.

### 5.2.4 FIT: IT Environment OS Compliance

FIT_PPC_EXP: IT Environment Protection Profile Compliance

FIT_PPC_EXP.1.1 The **environment** shall be compliant with the requirements of the Controlled Access Protection Profile or an Operating System Protection Profile at the Basic Level of Robustness or greater.

Application Note:

This requirement can be met by providing evidence (e.g., certificate) that the underlying operating system is compliant with the Controlled Access Protection Profile or with a protection profile at the Basic Level of Robustness or greater.

### 5.2.5 FTA: TOE Access

FTA_SSL.1: TSF-initiated session locking

Hierarchical to: No other components.

FTA_SSL.1.1 The **environment** shall lock a local interactive session after *a Web Server Administrator-specified time period of inactivity* by:

a) clearing or overwriting display devices, making the current contents unreadable.

b) disabling any activity of the user's data access/display devices other than unlocking the session.

FTA_SSL.1.2 The **environment** shall require the following events to occur prior to unlocking the session: *reauthentication by the administrative user*.

Dependencies: FIA_UAU.1 Timing of authentication

FTA_SSL.2: User-initiated locking

Hierarchical to: No other components

FTA_SSL.2.1 The **environment** shall allow user-initiated locking of the **Web Server Administrator's** own local interactive session by:

a) clearing or overwriting display devices, making the current contents unreadable.

b) disabling any activity of the user's data access/display devices other than unlocking the session.

FTA_SSL.2.2 The **environment** shall require the following events to occur prior to unlocking the session: *reauthentication by the Web Server Administrator*.

Dependencies: FIA_UAU.1 Timing of authentication

Application Note:

The interactive sessions in FTA_SSL.1 and FTA_SSL.2 are those of the local web server administrator. Non-administrators only have remote access to the TOE and the requirements for session locking levied on them are specified in FTA_SSL.3.

FTA_SSL.3/IN: TSF-initiated termination

Hierarchical to: No other components

FTA_SSL.3.1/IN The **environment** shall terminate a **remote** interactive session after a [*Web Server Administrator-configurable time interval of session inactivity*].

Dependencies: No dependencies

Application Note:

A remote interactive session applies to remote web server administrators.

FTA_TAB.1: Default TOE Access Banners

Hierarchical to: No other components

FTA_TAB.1.1 Before establishing a user session, the **environment** shall display an advisory warning message regarding unauthorized use of the TOE.

Dependencies: No dependencies

Application Note:

This has been restricted to administrative user sessions. Web user (and content provider, through HTTP) access has screens that are not under control of the web server, but under control of the content provider, and thus, outside the TSC.

## 5.3 Assurance Requirements

**Table 5-6: Assurance Requirements**

| Class | Identifier | Description |
|---|---|---|
| Configuration Management | ACM_CAP.2 | Configuration items |
| Delivery and operation | ADO_DEL.1 | Delivery procedures |
| | ADO_IGS.1 | Installation, generation, and start-up procedures |
| Development | ADV_FSP.1 | Informal functional specification |
| | ADV_HLD.1 | Descriptive high-level design |
| | ADV_RCR.1 | Informal correspondence demonstration |
| Guidance documents | AGD_ADM.1 | Administrator guidance |
| | AGD_USR.1 | User guidance |
| Tests | ATE_COV.1 | Evidence of coverage |
| | ATE_FUN.1 | Functional testing |

| Class | Identifier | Description |
|---|---|---|
| | ATE_IND.2 | Independent testing - sample |
| Vulnerability assessment | AVA_SOF.1 | Strength of TOE security function evaluation |
| | AVA_VLA.1 | Developer vulnerability analysis |

# 6.0 Rationale

This section describes the rationale for the Security Objectives and Security Functional Requirements as defined in Section 4.0 and Section 5.0, respectively. Additionally, this section describes the rationale for not satisfying all of the dependencies and the rationale for the strength of function (SOF) claim. Table 6 1 illustrates the mapping from Security Objectives to Threats and Policies.

## *6.1 Rationale for Organizational Security Policies*

This section provides a justification for the choice of Threats, Organizational Security Policies

**Table 6-1: Rationale for Organizational Security Policies**

| Policy | Objective | Rationale |
|---|---|---|
| P.CRY_APM | $O_{cm}$.COP_AMD | The Federal Government mandates the use of NIST approved cryptographic algorithms when cryptography is used to protected sensitive government information. $O_{cm}$.COP_AMD directly supports this government mandate. |
| P.CRY_VAL | $O_{cm}$.CMVP<br>$O_{cm}$.COP_SFT | The Federal Government mandates the use of FIPS 140-2 validation for all cryptographic functions used to protect sensitive government information. $O_{cm}$.CMVP directly supports this government mandate. $O_{cm}$.COP_SFT provides for startup testing of this module to ensure that it is functioning properly. |
| P.SYS_BNR | $O_{os}$.SYS_BNR | The Federal Government and most commercial companies are required to inform users that their actions may be monitored while they use computer systems. $O_{os}$.SYS_BNR directly supports this policy, P.SYS_BNR. |
| P.USR_ACC | $O_{ws}$.AUD_GEN<br>$O_{os}$.AUD_FUN<br>$O_{os}$.I&A | To hold users accountable for their actions, the users must be uniquely identified. $O_{os}$.I&A provides for the identification and authentication of users. $O_{os}$.AUD.FUN and $O_{ws}$.AUD.GEN provide for the collection, protection and review of events by administrators to enforce this policy. $O_{os}$.SYS_BNR provides for informing users that their actions are being monitored and that by using the system they consent to certain terms and conditions. |

P.CRY_APM Any cryptographic-based security must use NIST-approved algorithms.

Traces to: $O_{cm}$.COP_AMD

The Federal Government mandates the use of NIST approved cryptographic algorithms when cryptography is used to protected sensitive government information. $O_{cm}$.COP_AMD directly supports this government mandate.

P.CRY_VAL Any cryptographic-based security components used to protect sensitive information on U.S. Government computer must be FIPS 140-2 validated.

Traces to: $O_{cm}$.CMVP
$O_{cm}$.COP_SFT

The Federal Government mandates the use of FIPS 140-2 validation for all cryptographic functions used to protect sensitive government information. $O_{cm}$.CMVP directly supports this government mandate. $O_{cm}$.COP_SFT provides for startup testing of this module to ensure that it is functioning properly.

P.SYS_BNR Each computer system will display restrictions of use, legal agreements or any other appropriate information to which users consent by accessing the system.

Traces to: $O_{os}$.SYS_BNR

The Federal Government and most commercial companies are required to inform users that their actions may be monitored while they use computer systems. $O_{os}$.SYS_BNR directly supports this policy, P.SYS_BNR.

P.USR_ACC The users of the TOE will be held accountable for their actions within the TOE.

Traces to: $O_{ws}$.AUD_GEN
$O_{os}$.AUD_FUN
$O_{os}$.I&A
$O_{os}$.SYS_BNR

To hold users accountable for their actions, the users must be uniquely identified. $O_{os}$.I&A provides for the identification and authentication of users. $O_{os}$.AUD.FUN and $O_{ws}$.AUD.GEN provide for the collection, protection and review of events by administrators to enforce this policy. $O_{os}$.SYS_BNR provides for informing users that their actions are being monitored and that by using the system they consent to certain terms and conditions.

## 6.2 Rationale for Threats

This section presents each threat identified in the PP and identifies objective to mitigate the risk associated with the threat. A rationale is provided to justify the choice of objectives.

**Table 6-2: Rationale for Threats**

| Threat | Objective | Rationale |
|---|---|---|
| T.CAPTURE_TRAFFIC | $O_{WS}$.SSL_TLS | To prevent an adversary from intercepting and reconstructing controlled-access content, the TOE supports SSL/TLS. By using SSL/TLS, even if an adversary did capture the entire session between the web server and browser that session is encrypted with FIPS 140-2 validated cryptography making recovery extremely difficult, costly and time-consuming. |
| T.INVALID_URL | $O_{OS}$.AUD_FUN<br>$O_{OS}$.SYS_BNR<br>$O_{OS}$.SYS_PROT<br>$O_{WS}$.AUD_GEN<br>$O_{WS}$.SYS_PROT | The web server presents a simple and effective way to provide users with content. Static content is usually provided by returning the contents of a specified file to the user. To guard against poor web server implementation and provide a layered security design, the OS restricts the content the web server is able to access. The web server audits all security relevant events. The system banner explains to users (authorized or otherwise) that their actions are subject to monitoring[6]. |
| T.MASQUERADE | $O_{WS}$.SSL_TLS | Though similar to T.CAPTURE_TRAFFIC, T.MASQUERADE is focused on the replaying of captured traffic rather than the adversary actually reading any of the content. If the content being accessed were static content, then the attack makes little sense, since the response from the server would have been trivial to capture along with the request. The attack is then geared toward controlled-access content that is dynamically generated. The nature of SSL/TLS provides protection against replay attacks. |
| T.SERVER_MASQ | $O_{WS}$.SSL_TLS | Using digital certificates on the server with a trusted root authority guards against this attack. This functionality is supported through the use of SSL. However, SSL alone does not provide this ability; the user needs to verify the certificate chain. |

[6] No hackers have been successfully prosecuted if a banner warning them that the computer system is subject to monitoring is not present.

| Threat | Objective | Rationale |
|---|---|---|
| T.UNAUTHORIZED | $O_{ws}$.SSL_TLS | At stated in the TOE description, the TOE must serve controlled-access content using SSL/TLS. To access controlled-access content, the user must present a valid username/password or a personal digital certificate. Access to dynamic content may not be controlled by the web server. |
| $T_e$.PROVIDER_MASQ | $O_{os}$.AUD_FUN<br>$O_{os}$.I&A<br>$O_{os}$.SYS_PROT | The operating system is physically protected and access is only permitted by authorized personnel. $O_{os}$.I&A helps ensure that only authorized personnel can access the operating system or the TOE. Audit functions ($O_{os}$.AUD_FUN) provide accountability and $O_{os}$.SYS_PROT helps ensure that the content is appropriately protected. |
| $T_e$.REPLAY | $O_{os}$.SYS_PROT | The IT environment must handle this threat, though it is possible that the TOE counters it or helps counter it. There are several ways in which this threat can be countered including prohibiting administrators from connecting remotely to the TOE or using SSH or even I&A mechanisms which do not pass user ID and passwords in the clear. |
| $T_e$.TSF_BYPASS | $O_{os}$.SYS_PROT | The host computer system may present many interfaces through which an adversary may attempt an attack. Since the host computer system is physically protected, any attack must be launched through a network connection. For example, if content providers are to use FTP to upload content onto the web server, the FTP daemon becomes a potential point of attack for an adversary.<br><br>Once the adversary had gained access to the host OS, an attack could be launched on the web server or the OS in an attempt to gain access to controlled-access content. The host computer system must protect all exposed interfaces. It is highly advised that any unnecessary network daemons be disabled and all unnecessary applications be removed from the computer system. |

T.CAPTURE_TRAFFIC A web user may attempt to access non-public content by reading TCP/IP datagrams directly "off the wire" using a network traffic analyzer (e.g. "sniffer", packet analyzer, etc.) or a "man-in-the-middle" attack.

Traces to: $O_{WS}$.SSL_TLS

To prevent an adversary from intercepting and reconstructing controlled-access content, the TOE supports SSL/TLS. By using SSL/TLS, even if an adversary did capture the entire session between the web server and browser that session is encrypted with FIPS 140-2 validated cryptography making recovery extremely difficult, costly and time-consuming.

T.INVALID_URL A web user may attempt to create, modify or view controlled-access content, web server configuration files or OS specific files by entering an invalid URL or a URL specifically designed for this purpose.

Traces to: $O_{OS}$.AUD_FUN
$O_{OS}$.SYS_BNR
$O_{OS}$.SYS_PROT
$O_{WS}$.AUD_GEN
$O_{WS}$.SYS_PROT

The web server presents a simple and effective way to provide users with content. Static content is usually provided by returning the contents of a specified file to the user. To guard against poor web server implementation and provide a layered security design, the OS restricts the content the web server is able to access. The web server audits all security relevant events. The system banner explains to users (authorized or otherwise) that their actions are subject to monitoring[7].

T.MASQUERADE A user may masquerade or replay a previous session of another web user in order to access controlled content that would not normally be accessible.

Traces to: $O_{WS}$.SSL_TLS

Though similar to T.CAPTURE_TRAFFIC, T.MASQUERADE is focused on the replaying of captured traffic rather than the adversary actually reading any of the content. If the content being accessed were static content, then the attack makes little sense, since the response from the server would have been trivial to capture along with the request. The attack is then geared toward controlled-access content that is dynamically generated. The nature of SSL/TLS provides protection against replay attacks.

T.SERVER_MASQ A user may attempt to masquerade his web server as the legitimate web server to provide false or misleading content or capture user data.

Traces to: $O_{WS}$.SSL_TLS

[7] No hackers have been successfully prosecuted if a banner warning them that the computer system is subject to monitoring is not present.

Using digital certificates on the server with a trusted root authority guards against this attack. This functionality is supported through the use of SSL. However, SSL alone does not provide this ability; the user needs to verify the certificate chain.

T.UNAUTHORIZED A web user may request controlled-access content for which they are not authorized.

Traces to: $O_{ws}$.SSL_TLS

At stated in the TOE description, the TOE must serve controlled-access content using SSL/TLS. To access controlled-access content, the user must present a valid username/password or a personal digital certificate. Access to dynamic content may not be controlled by the web server.

$T_e$.PROVIDER_MASQ A user may attempt to create, modify or delete content that they are not authorized to by masquerading as the proper content provider.

Traces to: $O_{os}$.AUD_FUN<br>
$O_{os}$.I&A<br>
$O_{os}$.SYS_PROT

The operating system is physically protected and access is only permitted by authorized personnel. $O_{os}$.I&A helps ensure that only authorized personnel can access the operating system or the TOE. Audit functions ($O_{os}$.AUD_FUN) provide accountability and $O_{os}$.SYS_PROT helps ensure that the content is appropriately protected.

$T_e$.REPLAY A user may attempt to masquerade as the host OS administrator or web server administrator by capturing and replaying valid identification and authentication information.

Traces to: $O_{os}$.SYS_PROT

The IT environment must handle this threat, though it is possible that the TOE counters it or helps counter it. There are several ways in which this threat can be countered including prohibiting administrators from connecting remotely to the TOE or using SSH or even I&A mechanisms which do not pass user ID and passwords in the clear.

$T_e$.TSF_BYPASS A user may attempt to bypass the TSF to create, modify or delete controlled-access content, TSF data or other OS configuration files or the TOE by using non TOE interfaces of the host computer system.

Traces to: $O_{os}$.SYS_PROT

The host computer system may present many interfaces through which an adversary may attempt an attack. Since the host computer system is physically protected, any attack much be launched through a network connection. For example, if content providers are to use FTP to upload content onto the web server, the FTP daemon becomes a potential point of attack for an adversary.

Once the adversary had gained access to the host OS, an attack could be launched on the web server or the OS in an attempt to gain access to controlled-access content. The host computer system must protect all exposed interfaces. It is highly advised that any unnecessary network daemons be disabled and all unnecessary applications be removed from the computer system.

## 6.3 Rationale for Assumptions

This section presents each threat identified in the PP and identifies objective to mitigate the risk associated with the threat. A rationale is provided to justify the choice of objectives.

**Table 6-3: Rationale for Assumptions**

| Assumption | Objective | Rationale |
|---|---|---|
| A.ADM_GOOD<br>A.ADM_TRND<br>A.ADM_TRSTD | $O_E$.ADMIN | By having a process to hire the people that are trustworthy and qualified for the position, the assumption is met. |
| $A_{os}$.PHY_ACCES<br>$A_{os}$.PHY_PROT | $O_E$.PROT | By providing physical protection of the computer system, the possibility of tampering with the host computer system is eliminated. |
| $A_{ws}$.CPR_EAC<br>$A_{ws}$.CPR_GOOD<br>$A_{ws}$.CPR_TRND<br>$A_{ws}$.CPR_TRSTD | $O_E$.CON_PROV | By having a process to hire trustworthy, qualified people to produce and manage content, $A_{ws}$.CPR_EAC, $A_{ws}$.CPR_GOOD, $A_{ws}$.CPR_TRND, and $A_{ws}$.CPR_TRSTD are met. |
| $A_{ws}$.SYS_HIGH | $O_E$.SYS_HIGH | By providing an environment where all users can view all data served by the system (except the "need-to-know" allows for a system high environment where only discretionary access controls are necessary. |

A.ADM_GOOD Administrators will follow all published guidance.

Traces to: $O_E$.ADMIN

By following a defined process for the hiring and training of personnel, the assumption that administrators will follow all published guidance is reasonable.

A.ADM_TRND Administrators will be appropriately trained.

Traces to: $O_E$.ADMIN

By following a defined process for training personnel, the assumption that administrators are trained is reasonable.

A.ADM_TRSTD Administrators will not intentionally attempt to violate the TOE security policy or any environmental security policies necessary for the correct operation of the TOE.

Traces to: $O_E$.ADMIN

Since the personnel hired as administrators are trusted and well trained, these personnel know the importance of following stated security policies and thus, will follow all stated policy.

$A_{os}$.PHY_ACCES Physical access to the host computer system will be restricted to authorized personnel.

Traces to: $O_E$.PROT

The environment is providing physical protection and human access controls to protect the TOE. It is safe to assume that only authorized personnel will have access to the TOE.

$A_{os}$.PHY_PROT Physical protection of the host computer system will be commensurate with the value of that computer system and the data it contains.

Traces to: $O_E$.PROT

By providing a location for the TOE that is physically protected and provides access controls such that only authorized personnel are permitted access helps ensure that the $A_{os}$.PHY_PROT assumption is valid.

$A_{ws}$.CPR_EAC Content providers will establish access controls in accordance with the handling and dissemination procedures for that content.

Traces to: $O_E$.CON_PROV

Since a process exists to hire and train trusted and qualified content providers, it is reasonable to assume that these content providers will establish access controls in accordance with the handling and dissemination procedures for that content.

$A_{ws}$.CPR_GOOD Content providers will follow all published guidance.

Traces to: $O_E$.CON_PROV

All content providers are hired and trained through a well defined process that results in qualified and trusted personnel. Since these content providers are well trained, they know the importance of following security policy.

$A_{ws}$.CPR_TRND Content providers will be trained on the handling and dissemination procedures for the content for which they are responsible.

Traces to: $O_E$.CON_PROV

A process exists for hiring content providers that are trained on the proper handling and dissemination procedures for the content they are responsible for.

$A_{ws}$.CPR_TRSTD Content providers will not intentionally attempt to violate the TOE security policy or any environmental security policies necessary for the correct operation of the TOE.

Traces to: $O_E$.CON_PROV

Since content providers are trained and trusted, it is reasonable to assume that they will not intentionally attempt to violate any published security policies.

$A_{ws}$.SYS_HIGH All users with access to the host computer system possess proper personnel security clearance for all data contained on that system but only selected users or groups of users may obtain access to that data (e.g., based on a need-to-know).

Traces to: $O_E$.SYS_HIGH

By providing an environment where all users can view all data served by the system (except the "need-to-know" allows for a system high environment where only discretionary access controls are necessary.

### 6.4 Rationale for Security Functional Requirements

**Table 6-4: Rationale for SFRs**

| Objective | SFR | Rationale |
|---|---|---|
| $O_{cm}$.CMVP | FCS_BCM_EXP.1<br>FCS_CBP_EXP.1<br>FCS_CKM.1<br>FCS_CKM.4<br>FCS_CKM_EXP.1<br>FCS_CKM_EXP.2<br>FCS_CKM_EXP.3<br>FCS_CKM_EXP.4<br>FCS_CKM_EXP.5<br>FCS_CKM_EXP.6<br>FCS_COP.1(1)<br>FCS_COP.1(2)<br>FCS_COP.1(3)<br>FCS_COP.1(4)<br>FCS_KXP_EXP.1<br>FPT_RVM.1<br>FPT_SEP_EXP.1<br>FTA_SSL.3<br>FTP_ITC.1 | The selected FCS requirements were crafted to match the FIPS 140-2 specification to directly support the $O_{cm}$.CMVP objective. FPT_RVM and FPT_SEP_EXP ensure that the cryptographic module is not bypassed. |
| $O_{cm}$.COP_AMD | FCS_BCM_EXP.1 | FCS_BCM_EXP.1 is a direct translation of $O_{cm}$.COP_AMD. A TOE meeting FCS_BCM_EXP.1 will use NISP approved cryptographic mechanisms. |

| Objective | SFR | Rationale |
|---|---|---|
| $O_{cm}$.COP_SFT | FPT_TST.1/CR<br>FPT_TST_EXP.1/KG | FPT_TST.1/CR and FPT_TST_EXP.1/KG were designed to ensure that the cryptographic module performs a self-test to ensure that the module is working correctly for both key generation and encryption/decryption. |
| $O_{ws}$.AUD_GEN | FAU_GEN.1-NIAP-0410<br>FAU_GEN.2-NIAP-0410 | The FAU_GEN requirements directly support $O_{ws}$.AUD_GEN. |
| $O_{ws}$.SSL_TLS | FCS_BCM_EXP.1<br>FDP_UCT.1/WU<br>FDP_UIT.1/WU | A TOE meeting FCS_BCM_EXP.1, FDP_UCT.1/WU and FDP_UIT.1/WU will ensure that controlled-access content are protected using NISP approved and FIPS 140-2 validated cryptography. |
| $O_{ws}$.SYS_PROT | FDP_ACC.1/WU<br>FDP_ACF.1-NIAP-0407/WU<br>FDP_RIP.1<br>FIA_AFL.1-NIAP-0425<br>FIA_ATD.1<br>FIA_UAU.1<br>FIA_UID.1<br>FIA_USB.1-NIAP-0351<br>FMT_MOF.1<br>FMT_MSA.1<br>FMT_MSA.2<br>FMT_MSA.3-NIAP-0429<br>FMT_MTD.1<br>FMT_REV.1<br>FMT_SMF.1<br>FMT_SMR.1 | The FDP_ACC/ACF requirements provide for the protection of content while under the control of the TOE. Providing RIP also protects content from disclosure through poor coding techniques. The FIA family of requirements provide support for the FDP_ACC/ACF requirements. The FMT family of requirements provide for the secure management of the TOE. Together these requirements protect controlled-access content. |

# 7.0 Appendices

## 7.1 Appendix A – References

1) *Common Criteria for Information Technology Security Evaluation,* CCIB-98-031 Version 2.1, August 1999.

2) *Department of Defense Chief Information Officer Guidance and Policy Memorandum No. 6-8510, Guidance and Policy for the Department of Defense Global Information Grid Information Assurance (GIG)*, June 2000.

3) *Information Assurance Technical Framework*, Version 3.0, September 2000.

4) *Federal Information Processing Standard Publication (FIPS-PUB) 46-3, Data Encryption Standard (DES)*, October 1999.

5) *Federal Information Processing Standard Publication (FIPS-PUB) 140-2, Security Requirements for Cryptographic Modules*, May 25, 2001.

6) *Internet Engineering Task Force The TLS Protocol Version 1.0*, RFC 2246, January 1999

7) *Internet Engineering Task Force, Use of HMAC-SHA-1-96 within ESP and AH*, RFC 2404, November 1998.

8) *Internet Engineering Task Force, IP Encapsulating Security Payload (ESP), RFC 2406*, November 1998.

9) *Internet Engineering Task Force, Internet Key Exchange (IKE)*, RFC 2409, November 1998.

10) *Internet Engineering Task Force, ESP CBC-Mode Cipher Algorithms,* RFC 2451, November 1998.

11) *Internet Engineering Task Force, Hypertext Transfer Protocol -- HTTP/1.1,* RFC 2616, June 1999.

12) *Internet Engineering Task Force, HTTP Authentication: Basic and Digest Access Authentication*, RFC 2617, June 1999.

13) *Internet Engineering Task Force, Upgrading to TLS Within HTTP/1.1*, RFC 2817, May 2000.

14) *Internet Engineering Task Force, HTTP Over TLS*, RFC 2818, May 2000.

15) *Department of Defense Instruction, Information Assurance Implementation Draft No. 8500.bb*, September 2001.

16) *The AES Cipher Algorithm and Its Use with IPSec* <draft-ietf-ipsec-ciph-aes-cbc.03.txt>, Internet draft, November 2001.

17) *Federal Information Processing Standard Publication (FIPS-PUB) 197, Specification for the Advanced Encryption Standard (AES)*, November 26, 2001

## *7.2 Appendix B - Glossary*

This profile uses a number of terms in specific senses. The following sections provide definitions of the terms that are used in this PP.

**Accountability** — Property that allows activities in an IT system to be traced to the entity responsible for the activity.

**Assurance** — A measure of confidence that the security features of an IT system are sufficient to enforce its' security policy.

**Asymmetric Cryptographic System** — A system involving two related transformations; one determined by a public key (the public transformation), and another determined by a private key (the private transformation) with the property that it is computationally infeasible to determine the private transformation (or the private key) from knowledge of the public transformation (and the public key).

**Asymmetric Key** — The corresponding public/private key pair needed to determine the behavior of the public/private transformations that comprise an asymmetric cryptographic system.

**Authentication** — Security measure that verifies a claimed identity.

**Authentication data** — Information used to verify a claimed identity.

**Authorization** — Permission, granted by an entity authorized to do so, to perform functions and access data.

**Cryptographic Module** — The set of hardware, software, firmware, or some combination thereof that implements cryptographic logic or processes, including cryptographic algorithms, and is contained within the cryptographic boundary of the module.

**Cryptographic Module Security Policy** — A precise specification of the security rules under which a cryptographic module must operate, including the rules derived from the requirements of this PP and additional rules imposed by the vendor.

**Defense-in-Depth (DID)** — A security design strategy whereby layers of protection are utilized to establish an adequate security posture for an IT system.

**Discretionary Access Control (DAC)** — A means of restricting access to objects based on the identity of subjects and/or groups to which they belong. These controls are discretionary in the sense that a subject with a certain access permission is capable of passing that permission (perhaps indirectly) on to any other subject.

**Entity** — A subject, object, user, or another IT device, which interacts with TOE objects, data, or resources.

**External IT entity** — Any trusted Information Technology (IT) product or system, outside of the TOE, which may, in accordance with the TSP, perform an operation.

**Identity** — A representation (e.g., a string) uniquely identifying an authorized user. A common representation is the full or abbreviated name of that user or a pseudonym.

**Integrity** — A security policy pertaining to the corruption of data and TSF mechanisms.

**Message Authentication Code (MAC)** — A Message Authentication Code is a one-way hash computed from a message and some data. Its purpose is to detect if the message has been altered.

**Non-Repudiation** — A security policy pertaining to providing one or more of the following:

To the sender of data, proof of delivery to the intended recipient,

To the recipient of data, proof of the identity of the user who sent the data.

**Operating Environment** — The total environment in which a TOE operates. It includes the physical facility and any physical, procedural, administrative and personnel controls.

**Robustness** — A characterization of the strength of a security function, mechanism, service or solution, and the assurance (or confidence) that it is implemented and functioning correctly. There are three levels of robustness:

Basic: Security services and mechanisms that equate to good commercial practices. Basic robustness equates to EAL-2 plus; AMA (Maintenance of Assurance); ALC_FLR (Flaw Remediation), and AVA_MSU.1 (Misuse-Examination Guidance) as defined in CCIB-98-028, Part 3, Version 2.0

Medium: Security services and mechanisms that provide for layering of additional safeguards above good commercial practices. Medium robustness equates to EAL-4 plus; AMA (Maintenance of Assurance); ALC_FLR (Flaw Remediation); ADV_IMP.2; ADV_INT.1; ATE_DPT.2; and AVA_VLA.3 (Moderately Resistant Vulnerability Analysis) as defined in CCIB-98-028, Part 3, Version 2.0. If cryptographic functions are included in the TOE, then the PP should be augmented with AVA_CCA_EXP.2 as documented in the Protection Profile Medium Robustness Consistency Guidance.

High: Security services and mechanisms that provide the most stringent protection and rigorous security countermeasures.

**Secure State** — Condition in which all TOE security policies are enforced.

**Security attribute** — TSF data associated with subjects, objects, and users that is used for the enforcement of the TSP.

**Split key** — A variable that consists of two or more components that must be combined to form the operational key variable. The combining process excludes concatenation or interleaving of component variables.

**Subject** — An entity within the TSC that causes operations to be performed.

**Symmetric key** — A single, secret key used for both encryption and decryption in symmetric cryptographic algorithms.

**Threat** — Capabilities, intentions and attack methods of adversaries, or any circumstance or event, with the potential to violate the TOE security policy.

**Threat Agent** - Any human user or Information Technology (IT) product or system, which may attempt to violate the TSP and perform an unauthorized operation with the TOE.

**User** — Any entity (human user or external IT entity) outside the TOE that interacts with the TOE.

**Vulnerability** — A weakness that can be exploited to violate the TOE security policy.

### 7.3 Appendix C - Acronyms

The following abbreviations from the Common Criteria are used in this Protection Profile:

| | |
|---|---|
| ACL | Access Control List |
| AES | Advanced Encryption Standard |
| ATM | Asynchronous Transfer Method |
| CA | Certificate Authority |
| CAPP | Controlled Access Protection Profile |
| CC | Common Criteria for Information Technology Security Evaluation |
| CGI | Common Gateway Interface |
| DES | Data Encryption Standard |
| DMZ | Demilitarized zone |
| DoD | Department of Defense |
| EAL | Evaluation Assurance Level |
| ESP | Encapsulating Security Payload |
| FIPS PUB | Federal Information Processing Standard Publication |
| FTP | File Transfer Protocol |
| GIG | Global Information Grid |
| HTML | Hypertext Markup Language |
| HTTP | Hypertext Transfer Protocol |
| HTTPS | HTTP with a Secure Socket Layer (SSL) |
| I&A | Identification and Authentication |
| IAFT | Information Assurance Technical Framework |
| IATF | Information Assurance Technical Framework |
| ICMP | Internet Control Message Protocol |
| IETF | Internet Engineering Task Force |
| IKE | Internet Key Exchange |
| IP | Internet Protocol |
| IT | Information Technology |
| N/A | Not Applicable |
| NIAP | National Information Assurance Partnership |
| NIST | National Institute of Standards and Technology |
| NSA | National Security Agency |
| PKI | Public Key Infrastructure |
| PP | Protection Profile |
| RNG | Random Number Generator |
| SF | Security Function |
| SFP | Security Function Policy |
| SOF | Strength of Function |
| SSL | Secure Socket Layer |
| ST | Security Target |
| TCP | Transmission Control Protocol |
| TFTP | Trivial File Transfer Protocol |
| TLS | Transport Layer Security |
| TOE | Target of Evaluation |

| | |
|---|---|
| TSC | TSF Scope of Control |
| TSE | TOE Security Environment |
| TSF | TOE Security Function |
| TSP | TOE Security Policy |
| URI | Universal Resource Identifier |
| URL | Uniform Resource Locator |
| WWW | World Wide Web |

## *7.4 Appendix D - Robustness Environment Characterization*

### 7.4.1 General Environmental Characterization

In trying to specify the environments in which TOEs with various levels of robustness are appropriate, it is useful to first discuss the two defining factors that characterize that environment: **value of the resources** and **authorization of the entities** to those resources.

In general terms, the environment for a TOE can be characterized by the authorization (or lack of authorization) the least trustworthy entity has with respect to the highest value of TOE resources (i.e. the TOE itself and all of the data processed by the TOE).

Note that there are an infinite number of combinations of entity authorization and value of resources; this conceptually “makes sense” because there are an infinite number of potential environments, depending on how the resources are valued by the organization, and the variety of authorizations the organization defines for the associated entities. In section 1.2.2, these two environmental factors will be related to the robustness required for selection of an appropriate TOE.

### 7.4.2 VALUE OF RESOURCES

Value of the resources associated with the TOE includes the data being processed or used by the TOE, as well as the TOE itself (for example, a real-time control processor). “Value” is assigned by the using organization. For example, in the DoD low-value data might be equivalent to data marked “FOUO”, while high-value data may be those classified Top Secret. In a commercial enterprise, low-value data might be the internal organizational structure as captured in the corporate on-line phone book, while high-value data might be corporate research results for the next generation product. Note that when considering the value of the data one must also consider the value of data or resources that are accessible through exploitation of the TOE. For example, a firewall may have “low value” data itself, but it might protect an enclave with high value data. If the firewall was being depended upon to protect the high value data, then it must be treated as a high-value-data TOE.

### 7.4.3 AUTHORIZATION OF ENTITIES

Authorization that entities (users, administrators, other IT systems) have with respect to the TOE (and thus the resources of that TOE, including the TOE itself) is an abstract concept reflecting a combination of the trustworthiness of an entity and the access and privileges granted to that entity with respect to the resources of the TOE. For instance, entities that have total authorization to all data on the TOE are at one end of this spectrum; these entities may have privileges that allow them to read, write, and modify anything on the TOE, including all TSF data. Entities at the other end of the spectrum are those that are authorized to few or no TOE resources. For example, in the case of a router, non-administrative entities may have their packets routed by the TOE, but that is the extent of their authorization to the TOE's resources. In the case of an OS, an entity may not be allowed to log on to the TOE at all (that is, they are not valid users listed in the OS's user database).

It is important to note that authorization does not refer to the access that the entities actually have to the TOE or its data. For example, suppose the owner of the system determines that no one other than employees was authorized to certain data on a TOE, yet they connect the TOE to the Internet. There are millions of entities that are not authorized to the data (because they are not employees), but they actually have connectivity to the TOE through the Internet and thus can attempt to access the TOE and its associated resources.

Entities are characterized according to the value of resources to which they are authorized; the extent of their authorization is implicitly a measure of how trustworthy the entity is with respect to compromise of the data (that is, compromise of any of the applicable security policies; e.g., confidentiality, integrity, availability). In other words, in this model the greater the extent of an entity's authorization, the more trustworthy (with respect to applicable policies) that entity is.

### 7.4.4 SELECTION OF APPROPRIATE ROBUSTNESS LEVELS

Robustness is a characteristic of a TOE defining how well it can protect itself and its resources; a more robust TOE is better able to protect itself. This section relates the defining factors of IT environments, authorization, and value of resources to the selection of appropriate robustness levels.

When assessing any environment with respect to Information Assurance the critical point to consider is the likelihood of an attempted security policy compromise, which was characterized in the previous section in terms of entity authorization and resource value. As previously mentioned, robustness is a characteristic of a TOE that reflects the extent to which a TOE can protect itself and its resources. It follows that as the likelihood of an attempted resource compromise increases, the robustness of an appropriate TOE should also increase.

It is critical to note that several combinations of the environmental factors will result in environments in which the likelihood of an attempted security policy compromise is similar. Consider the following two cases:

The first case is a TOE that processes only low-value data. Although the organization has stated that only its employees are authorized to log on to the system and access the data, the system is connected to the Internet to allow authorized employees to access the system from home. In this case, the least trusted entities would be unauthorized entities (e.g. non-employees) exposed to the TOE because of the Internet connectivity. However, since only low-value data are being processed, the likelihood that unauthorized entities would find it worth their while to attempt to compromise the data on the system is low and selection of a basic robustness TOE would be appropriate.

The second case is a TOE that processes high-value (e.g., classified) information. The organization requires that the TOE be stand-alone, and that every user with physical and logical access to the TOE undergo an investigation so that they are authorized to the highest value data on the TOE. Because of the extensive checks done during this investigation, the organization is assured that only highly trusted users are authorized to use the TOE. In this case, even though high value information is being processed, it is unlikely that a compromise of that data will be attempted because of the authorization and trustworthiness of the users and once again, selection of a basic robustness TOE would be appropriate.

The preceding examples demonstrated that it is possible for radically different combinations of entity authorization/resource values to result in a similar likelihood of an attempted compromise. As mentioned earlier, the robustness of a system is an indication of the protection being provided to counter compromise attempts. Therefore, a basic robustness system should be sufficient to counter compromise attempts where the likelihood of an attempted compromise is low. The following chart depicts the "universe" of environments characterized by the two factors discussed in the previous section: on one axis is the authorization defined for the least trustworthy entity, and on the other axis is the highest value of resources associated with the TOE.

As depicted in the following figure, the robustness of the TOEs required in each environment steadily increases as one goes from the upper left of the chart to the lower right; this corresponds to the need to counter increasingly likely attack attempts by the least trustworthy entities in the environment. Note that the shading of the chart is intended to reflect- the notion that different environments engender similar levels of "likelihood of attempted compromise", signified by a similar color. Further, the delineations between such environments are not stark, but rather are finely grained and gradual.

While it would be possible to create many different "levels of robustness" at small intervals along the "Increasing Robustness Requirements" line to counter the increasing likelihood of attempted compromise due to those attacks, it would not be practical nor particularly useful. Instead, in order to implement the robustness strategy where there are only three robustness levels: Basic, Medium, and High, the graph is divided into three sections, with each section corresponding to a set of environments where the likelihood of attempted compromise is roughly similar. This is graphically depicted in the following chart.

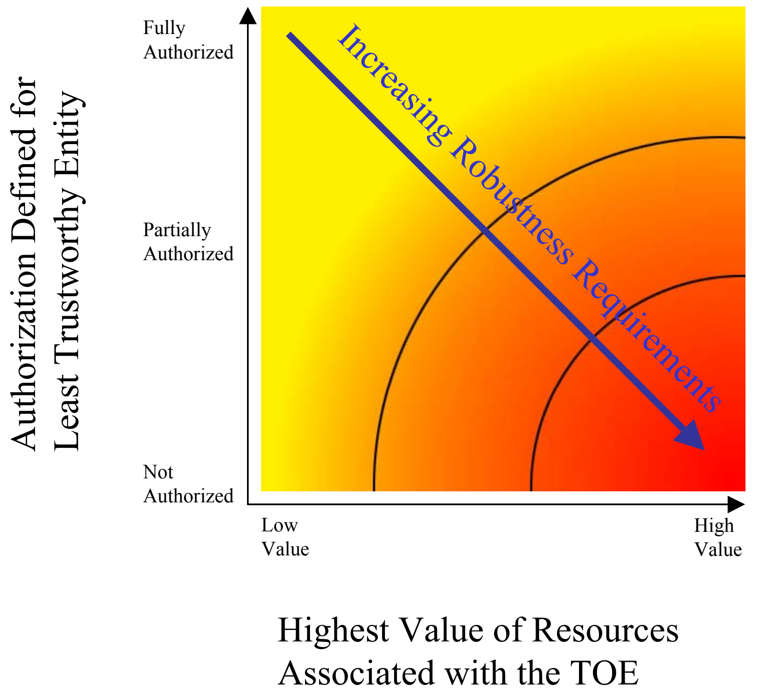
Authorization Defined for Least Trustworthy Entity
Fully Authorized
Partially Authorized
Not Authorized
Increasing Robustness Requirements
Low Value
High Value
Highest Value of Resources Associated with the TOE

**Figure 7-1: Increasing Robustness Requirements**

In this second representation of environments and the robustness plane below, the "dots" represent given instantiations of environments; like-colored dots define environments with a similar likelihood of attempted compromise. Correspondingly, a TOE with a given robustness should provide sufficient protection for environments characterized by like-colored dots.

In choosing the appropriateness of a given robustness level TOE PP for an environment, then, the user must first consider the lowest authorization for an entity as well as the highest value of the resources in that environment. This should result in a "point" in the chart above, corresponding to the likelihood that that entity will attempt to compromise the most valuable resource in the environment. The appropriate robustness level for the specified TOE to counter this likelihood can then be chosen. The difficult part of this activity is differentiating the authorization of various entities, as well as determining the relative values of resources; (e.g., what constitutes "low value" data vs. "medium value" data). Because every organization will be different, a rigorous definition is not possible. In section 3.1, the targeted threat level for a Basic robustness TOE is characterized. This information is provided to help organizations using this PP -ensure that the functional requirements specified by this Basic robustness PP are appropriate for their intended application of a compliant TOE.

In this second representation of environments and the robustness plane below, the "dots" represent given instantiations of environments; like-colored dots define environments with a similar likelihood of attempted compromise. Correspondingly, a TOE with a given robustness should provide sufficient protection for environments characterized by like-colored dots. In choosing the appropriateness of a given robustness level TOE PP for an environment, then, the user must first consider the lowest authorization for an entity as well as the highest value of the resources in that environment. This should result in a "point" in the chart above, corresponding to the likelihood that that entity will attempt to compromise the most valuable resource in the environment. The appropriate robustness level for the specified TOE to counter this likelihood can then be chosen.

The difficult part of this activity is differentiating the authorization of various entities, as well as determining the relative values of resources; (e.g., what constitutes “low value” data vs. “medium value” data). Because every organization will be different, a rigorous definition is not possible.

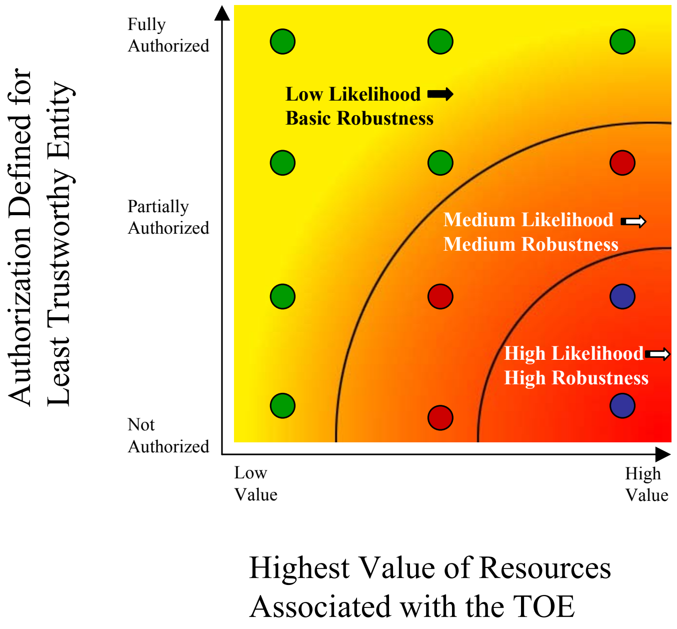
Authorization Defined for
Least Trustworthy Entity
Fully Authorized
Partially Authorized
Not Authorized
Low Likelihood ➡ Basic Robustness
Medium Likelihood ⇨ Medium Robustness
High Likelihood ⇨ High Robustness
Low Value
High Value
Highest Value of Resources
Associated with the TOE

**Figure 7-2: Basic, Medium and High Robustness**